

Final Study Data Analysis

April Kim, Jennifer Podracky, Saurav Datta

```
library(ggplot2)
library(data.table)
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
```

```
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
##      as.Date, as.Date.numeric
```

```
library(pwr)
library(lsr)
library(cobalt)
library(stringr)
library(AER)
```

```
## Loading required package: car
```

```
## Loading required package: carData
```

```
## Loading required package: sandwich
```

```
## Loading required package: survival
```

```
library(stargazer)
```

```
##
```

```
## Please cite as:
```

```
## Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.
```

```
## R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
```

```
library(pander)
```

Read in data and reformat

```

assigned_treatment_seq <- data.frame(seq_id = c(1,2,3,4,5,6),
                                     day1 = c(0,0,1,1,2,2),
                                     day2 = c(1,2,0,2,0,1),
                                     day3 = c(2,1,2,0,1,0))
d2 <- fread("241 Participant List - Final Study Results - 20181215.csv", na.strings=c("", "NA"))
d2[UserId == 65,]$Q10 <- "In person"
d2[UserId == 13,]$Q6 <- "Through digital means"
# stringsAsFactors = F)
names(d2) <- str_replace_all(names(d2), c(" " = ".", ", " = "" ))
# subset d2 for those who responded (Submitted.Data = 1)
d2 <- d2[Submitted.Data == 1]

# Not applicable = 0
# Through digital means = 1
# In person = 2
# Both in person and through digital means = 3

d2 <- d2[, .(userId = UserId,
             treatment_seq = as.integer(Treatment.Seq),
             day1_treatment = as.integer(as.character(factor(Q6, levels = c('Not applicable', 'In person',
                                                                           'Through digital means'),
                                                                           labels = c(0, 2, 1)))),
             day2_treatment = as.integer(as.character(factor(Q10, levels = c('Not applicable', 'In person',
                                                                           'Through digital means',
                                                                           'Both in person and through digital means'),
                                                                           labels = c(0, 2, 1, 3)))),
             day3_treatment = as.integer(as.character(factor(Q14, levels = c('Not applicable', 'In person',
                                                                           'Through digital means',
                                                                           'Both in person and through digital means'),
                                                                           labels = c(0, 2, 1, 3)))),
             day1_steps = as.numeric(gsub("\\\\", "", Q7)),
             day2_steps = as.numeric(gsub("\\\\", "", Q11)),
             day3_steps = as.numeric(gsub("\\\\", "", Q15)),
             age_range = as.integer(as.character(factor(Age, levels = c('18 - 24',
                                                                           '25 - 34',
                                                                           '35 - 44',
                                                                           '45 - 54',
                                                                           '55 - 64',
                                                                           '65+'),
                                                                           labels = c(0, 1, 2, 3, 4, 5)))),
             # gender = factor(Gender),
             gender = as.integer(as.character(factor(Gender, levels = c('Male', 'Female', 'Gender non-conforming'),
                                                                           labels = c(0, 1, 2)))),
             lives_with_others = as.integer(as.character(factor(Living.Situation, levels = c('Alone', 'With others'),
                                                                           labels = c(0, 1)))),
             # know_us = factor(Q17),
             know_us = as.integer(as.character(factor(Q17, levels = c('No', 'Yes'),
                                                                           labels = c(0, 1)))),
             location_lat = as.double(LocationLatitude),
             location_long = as.double(LocationLongitude)
)]

```

```
## Warning in eval(jsub, SEnv, parent.frame()): NAs introduced by coercion
```

```
## Warning in eval(jsub, SDenv, parent.frame()): NAs introduced by coercion
```

```
## Warning in eval(jsub, SDenv, parent.frame()): NAs introduced by coercion
```

```
d2$gender[is.na(d2$gender)] <- 2
d2$age_range[is.na(d2$age_range)] <- 6
d2$lives_with_others[is.na(d2$lives_with_others)] <- 2
d2$know_us[is.na(d2$know_us)] <- 2

head(d2, 5)
```

```
##      userId treatment_seq day1_treatment day2_treatment day3_treatment
## 1:      82             6             0             1             0
## 2:      57             3             1             0             2
## 3:      69             3             1             0             2
## 4:      85             3             1             0             2
## 5:      66             4             1             2             0
##      day1_steps day2_steps day3_steps age_range gender lives_with_others
## 1:      NA      5040      3788          1      0             1
## 2:    21290    13959    13717          0      0             1
## 3:    6343     3247    10198          1      0             1
## 4:   13624     5406     7851          1      1             1
## 5:    7016     1211     5717          0      0             1
##      know_us location_lat location_long
## 1:      1      41.89250      -87.7895
## 2:      1      37.75101      -97.8220
## 3:      1      40.37070      -74.0084
## 4:      1      42.41730      -71.1087
## 5:      1      42.35760      -71.0514
```

```
#Covariate Balance Check
```

```
bal.tab(treatment_seq ~ factor(gender) + factor(age_range) + factor(lives_with_others) + factor(know_us),
        data = d2)
```

```
## Balance Measures
```

```
##                                     Type Corr.Un
## factor(gender)_0                   Binary -0.0770
## factor(gender)_1                   Binary  0.1053
## factor(gender)_2                   Binary -0.1007
## factor(age_range)_0                Binary  0.0138
## factor(age_range)_1                Binary -0.0633
## factor(age_range)_2                Binary  0.2684
## factor(age_range)_3                Binary -0.0503
## factor(age_range)_4                Binary  0.0677
## factor(age_range)_5                Binary -0.1438
## factor(lives_with_others)_0        Binary -0.0292
## factor(lives_with_others)_1        Binary -0.0094
## factor(lives_with_others)_2        Binary  0.0677
## factor(know_us)_1                  Binary  0.0426
## location_lat                       Contin.  0.1095
## location_long                      Contin.  0.0512
##
```

```
## Sample sizes
##      Total
## All    51
```

```
cov_check <- lm(treatment_seq ~ factor(gender) + factor(age_range) + factor(lives_with_others) + factor
               data = d2)
summary(cov_check)
```

```
##
## Call:
## lm(formula = treatment_seq ~ factor(gender) + factor(age_range) +
##     factor(lives_with_others) + factor(know_us) + location_lat +
##     location_long, data = d2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.6981 -0.9913  0.0000  0.9265  2.9062
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -6.17461     6.18816  -0.998   0.3247
## factor(gender)1       0.37850     0.57242   0.661   0.5125
## factor(gender)2      -1.89910     2.02685  -0.937   0.3547
## factor(age_range)1    -0.19090     0.77885  -0.245   0.8077
## factor(age_range)2     2.25170     1.29109   1.744   0.0892
## factor(age_range)3     0.41468     1.07541   0.386   0.7019
## factor(age_range)4     0.77973     1.96081   0.398   0.6931
## factor(age_range)5    -1.18844     1.44306  -0.824   0.4153
## factor(lives_with_others)1  0.52266     1.15299   0.453   0.6529
## factor(lives_with_others)2  1.27148     2.20112   0.578   0.5669
## factor(know_us)1        0.98364     1.00578   0.978   0.3343
## location_lat          0.15597     0.11227   1.389   0.1729
## location_long        -0.01893     0.01889  -1.002   0.3225
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.768 on 38 degrees of freedom
## Multiple R-squared:  0.1755, Adjusted R-squared:  -0.08485
## F-statistic: 0.6741 on 12 and 38 DF,  p-value: 0.7647
```

Checking for ordering/priming effect AND adding non-compliant but okay users

Is previous day's treatment highly predictive of how many steps are taken today?

```
'%!in%' <- function(x,y){('%in%'(x,y))
# n = 51
df1 <- d2

# remove subjects/rows who were non-compliant (n = 2)
# n = 49
df1 <- df1[rowSums(is.na(df1[,c(6:8)])) != ncol(df1[,c(6:8)]), ]
```

```
head(df1, 5)
```

```
##      userId treatment_seq day1_treatment day2_treatment day3_treatment
## 1:      82           6           0           1           0
## 2:      57           3           1           0           2
## 3:      69           3           1           0           2
## 4:      85           3           1           0           2
## 5:      66           4           1           2           0
##      day1_steps day2_steps day3_steps age_range gender lives_with_others
## 1:      NA      5040      3788           1      0           1
## 2:    21290    13959    13717           0      0           1
## 3:    6343     3247    10198           1      0           1
## 4:   13624     5406     7851           1      1           1
## 5:    7016     1211     5717           0      0           1
##      know_us location_lat location_long
## 1:      1      41.89250      -87.7895
## 2:      1      37.75101      -97.8220
## 3:      1      40.37070      -74.0084
## 4:      1      42.41730      -71.1087
## 5:      1      42.35760      -71.0514
```

```
# n = 30
```

```
d_followed_treatment_sequence <- rbindlist(list(subset(df1, treatment_seq == 1 & df1$day1_treatment == assigned_treatment_seq[1]
& df1$day2_treatment == assigned_treatment_seq[1]
& df1$day3_treatment == assigned_treatment_seq[1]
subset(df1, treatment_seq == 2 & df1$day1_treatment == assigned_treatment_seq[2]
& df1$day2_treatment == assigned_treatment_seq[2]
& df1$day3_treatment == assigned_treatment_seq[2]
subset(df1, treatment_seq == 3 & df1$day1_treatment == assigned_treatment_seq[3]
& df1$day2_treatment == assigned_treatment_seq[3]
& df1$day3_treatment == assigned_treatment_seq[3]
subset(df1, treatment_seq == 4 & df1$day1_treatment == assigned_treatment_seq[4]
& df1$day2_treatment == assigned_treatment_seq[4]
& df1$day3_treatment == assigned_treatment_seq[4]
subset(df1, treatment_seq == 5 & df1$day1_treatment == assigned_treatment_seq[5]
& df1$day2_treatment == assigned_treatment_seq[5]
& df1$day3_treatment == assigned_treatment_seq[5]
subset(df1, treatment_seq == 6 & df1$day1_treatment == assigned_treatment_seq[6]
& df1$day2_treatment == assigned_treatment_seq[6]
& df1$day3_treatment == assigned_treatment_seq[6]
)))
```

```
# n = 19
```

```
d_not_followed_treatment_sequence <- subset(df1, userId %!in% d_followed_treatment_sequence$userId)

d_not_followed_but_ok <- subset(d_not_followed_treatment_sequence, d_not_followed_treatment_sequence$day1_treatment != d_not_followed_treatment_sequence$day1_treatment |
d_not_followed_treatment_sequence$day2_treatment != d_not_followed_treatment_sequence$day2_treatment |
d_not_followed_treatment_sequence$day3_treatment != d_not_followed_treatment_sequence$day3_treatment)

na.omit(d_not_followed_but_ok)
```

```
##      userId treatment_seq day1_treatment day2_treatment day3_treatment
```

```
## 1:      3      3      1      2      0
## 2:     73      5      2      1      0
## 3:     75      5      2      1      0
##   day1_steps day2_steps day3_steps age_range gender lives_with_others
## 1:      7000      5000      6000         1      1              1
## 2:      6050      5671      3251         1      0              1
## 3:     10422      5187      9696         2      0              1
##   know_us location_lat location_long
## 1:      1      48.2804      11.5768
## 2:      1      42.3576     -71.0514
## 3:      1      42.3576     -71.0514
```

```
d_not_followed_no_NA <- subset(d_not_followed_treatment_sequence, userId %!in% d_not_followed_but_ok$userId)
# n = 15
d_not_followed_no_NA <- na.omit(d_not_followed_no_NA)

# n = 33
df <- rbind(d_followed_treatment_sequence, d_not_followed_but_ok)
# n = 48
df2 <- rbind(d_followed_treatment_sequence, d_not_followed_but_ok, d_not_followed_no_NA)

# day 3 steps using day 1 and 2 treatment on complied + people who followed within subject design
m1 <- lm(day3_steps ~ day1_treatment + day2_treatment, df)
summary(m1)
```

```
##
## Call:
## lm(formula = day3_steps ~ day1_treatment + day2_treatment, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -7878.7 -1605.2   -79.7   1887.3   6165.3
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    8615.6     1381.1   6.238  8.3e-07 ***
## day1_treatment    -736.9       804.1  -0.916    0.367
## day2_treatment   -957.0       804.1  -1.190    0.244
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3296 on 29 degrees of freedom
## (1 observation deleted due to missingness)
## Multiple R-squared:  0.06136,    Adjusted R-squared:  -0.003372
## F-statistic: 0.9479 on 2 and 29 DF,  p-value: 0.3992
```

```
# ATE (standard error)
print(paste0("Estimated effect of day1 treatment: ", signif(m1$coefficients[2], 3),
" (", signif(coef(summary(m1))[2,2], 3), ")"))
```

```
## [1] "Estimated effect of day1 treatment: -737 (804)"
```

```
print(paste0("Estimated effect of day2 treatment: ", signif(m1$coefficients[3], 3),
" (", signif(coef(summary(m1))[3,2], 3), ")"))
```

```
## [1] "Estimated effect of day2 treatment: -957 (804)"
```

```
# include days1,2 steps as covariates to understand
# subjects' step counts have as a function of
# treatment against what they would typically do
m2 <- lm(day3_steps ~ day1_treatment + day2_treatment + day1_steps + day2_steps, df)
summary(m2)
```

```
##
## Call:
## lm(formula = day3_steps ~ day1_treatment + day2_treatment + day1_steps +
##     day2_steps, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -8317.4  -898.4    88.7  1156.5  5167.1
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2534.372   2161.0040   1.173   0.2511
## day1_treatment -392.4212    702.2724  -0.559   0.5809
## day2_treatment  -8.3389    746.4587  -0.011   0.9912
## day1_steps      0.3441     0.1553    2.216   0.0353 *
## day2_steps      0.2922     0.2029    1.440   0.1614
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2845 on 27 degrees of freedom
## (1 observation deleted due to missingness)
## Multiple R-squared:  0.3489, Adjusted R-squared:  0.2524
## F-statistic: 3.617 on 4 and 27 DF,  p-value: 0.01741
```

```
print(paste0("Estimated effect of day1 treatment: ", signif(m2$coefficients[2], 3),
" (", signif(coef(summary(m2))[2,2], 3), ")"))
```

```
## [1] "Estimated effect of day1 treatment: -392 (702)"
```

```
print(paste0("Estimated effect of day2 treatment: ", signif(m2$coefficients[3], 3),
" (", signif(coef(summary(m2))[3,2], 3), ")"))
```

```
## [1] "Estimated effect of day2 treatment: -8.34 (746)"
```

```
# day 3 steps using day 1 and 2 treatment on complied + people who followed within subject design + res
m1 <- lm(day3_steps ~ day1_treatment + day2_treatment, df2)
summary(m1)
```

```
##
## Call:
## lm(formula = day3_steps ~ day1_treatment + day2_treatment, data = df2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -7168.1 -2208.5  -428.6  1439.1  8449.1
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    7645.4     932.3    8.200 2.08e-10 ***
## day1_treatment  -477.2     679.0   -0.703   0.486
## day2_treatment  -369.8     600.9   -0.615   0.542
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3440 on 44 degrees of freedom
## (1 observation deleted due to missingness)
## Multiple R-squared:  0.01807,    Adjusted R-squared:  -0.02657
## F-statistic: 0.4047 on 2 and 44 DF,  p-value: 0.6696

# ATE (standard error)
print(paste0("Estimated effect of day1 treatment: ", signif(m1$coefficients[2], 3),
            " (", signif(coef(summary(m1))[2,2], 3), ")"))

## [1] "Estimated effect of day1 treatment: -477 (679)"

print(paste0("Estimated effect of day2 treatment: ", signif(m1$coefficients[3], 3),
            " (", signif(coef(summary(m1))[3,2], 3), ")"))

## [1] "Estimated effect of day2 treatment: -370 (601)"

# include days1,2 steps as covariates to understand
# subjects' step counts have as a function of
# treatment against what they would typically do
m2 <- lm(day3_steps ~ day1_treatment + day2_treatment + day1_steps + day2_steps, df2)
summary(m2)

##
## Call:
## lm(formula = day3_steps ~ day1_treatment + day2_treatment + day1_steps +
##      day2_steps, data = df2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -8183.6 -1375.5    61.2  1536.6  5057.0
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2602.4071  1128.7513   2.306  0.02614 *
## day1_treatment -484.8396   517.7869  -0.936  0.35444
## day2_treatment -296.7856   460.9784  -0.644  0.52319
```



```
## day1_steps      0.2442      0.1230      1.985      0.05373 .
## day2_steps      0.4542      0.1341      3.386      0.00155 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2623 on 42 degrees of freedom
## (1 observation deleted due to missingness)
## Multiple R-squared:  0.455, Adjusted R-squared:  0.4031
## F-statistic: 8.766 on 4 and 42 DF, p-value: 3.076e-05
```

```
print(paste0("Estimated effect of day1 treatment: ", signif(m2$coefficients[2], 3),
            " (", signif(coef(summary(m2))[2,2], 3), ")"))
```

```
## [1] "Estimated effect of day1 treatment: -485 (518)"
```

```
print(paste0("Estimated effect of day2 treatment: ", signif(m2$coefficients[3], 3),
            " (", signif(coef(summary(m2))[3,2], 3), ")"))
```

```
## [1] "Estimated effect of day2 treatment: -297 (461)"
```

We do not see that the previous days' treatment assignments to predict the last day's step count is highly predictive and significant, which is super for us!

```
stargazer(m1, m2,
          dep.var.labels=c("Steps - Day 3"),
          covariate.labels=c("Treatment - Day 1", "Treatment - Day 2", "Steps - Day 1", "Steps - Day 2"),
          omit.stat=c("all"))
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu
 % Date and time: Thu, Dec 20, 2018 - 21:44:45

Condense treatment sequence to 1 treatment

```
df1.1 <- df[, -c(4,5,7,8)]
df2.1 <- df[, -c(3,5,6,8)]
df3.1 <- df[, -c(3,4,6,7)]
names(df1.1)[names(df1.1) == "day1_treatment"] = "treatment"
names(df1.1)[names(df1.1) == "day1_steps"] = "steps"
names(df2.1)[names(df2.1) == "day2_treatment"] = "treatment"
names(df2.1)[names(df2.1) == "day2_steps"] = "steps"
names(df3.1)[names(df3.1) == "day3_treatment"] = "treatment"
names(df3.1)[names(df3.1) == "day3_steps"] = "steps"
d <- rbind(df1.1, df2.1, df3.1)
# combine digital and in person treatment as one
d$treatment2 <- ifelse(d$treatment == 0, 0, 1)
d$outcome <- ifelse(d$steps > 5000, 1, 0)
head(d, 5)
```

Table 1:

	<i>Dependent variable:</i>	
	Steps - Day 3	
	(1)	(2)
Treatment - Day 1	-477.240 (679.024)	-484.840 (517.787)
Treatment - Day 2	-369.761 (600.936)	-296.786 (460.978)
Steps - Day 1		0.244* (0.123)
Steps - Day 2		0.454*** (0.134)
Constant	7,645.381*** (932.308)	2,602.407** (1,128.751)
<i>Note:</i> *p<0.1; **p<0.05; ***p<0.01		

```
##      userId treatment_seq treatment steps age_range gender lives_with_others
## 1:      28             1         0 13929         0       0             1
## 2:      56             1         0  5368         1       1             1
## 3:      25             1         0  5802         1       0             1
## 4:      22             1         0  5689         3       0             1
## 5:      86             1         0  5868         1       0             1
##      know_us location_lat location_long treatment2 outcome
## 1:         1    36.05251    -79.1077         0         1
## 2:         1    42.35760    -71.0514         0         1
## 3:         1    42.37700    -71.1256         0         1
## 4:         1    42.35760    -71.0514         0         1
## 5:         1    42.61240    -83.0345         0         1
```

Make some pretty plots to show distribution, populatin etc.

```
# population that actually responded to data collection survey
require(gridExtra)
```

```
## Loading required package: gridExtra
```

```
d.gender <- d[, c("gender", "treatment2")]
p_gender <- ggplot(d.gender, aes(x=gender, fill = factor(treatment2))) +
  geom_bar(stat="count", position=position_dodge()) +
  theme_minimal() + theme(legend.position="right") +
  xlab("") + ylab("") + ggtitle("Gender") +
  guides(fill = guide_legend(title = "Assignment")) +
```

```

scale_fill_discrete(labels = c("Control", "Treatment")) +
scale_x_continuous(breaks = c(0, 1, 2),
                   labels = c('Male', 'Female', 'Gender\n non-conforming'))

p_gender_no_legend <- ggplot(d.gender, aes(x=gender, fill = factor(treatment2))) +
  geom_bar(stat="count", position=position_dodge()) +
  theme_minimal() + theme(legend.position="none") +
  xlab("") + ylab("") + ggtitle("Gender") +
  # guides(fill = guide_legend(title = "Assignment")) +
  # scale_fill_discrete(labels = c("Control", "Treatment")) +
  scale_x_continuous(breaks = c(0, 1, 2),
                    labels = c('Male', 'Female', 'Gender\n non-conforming'))

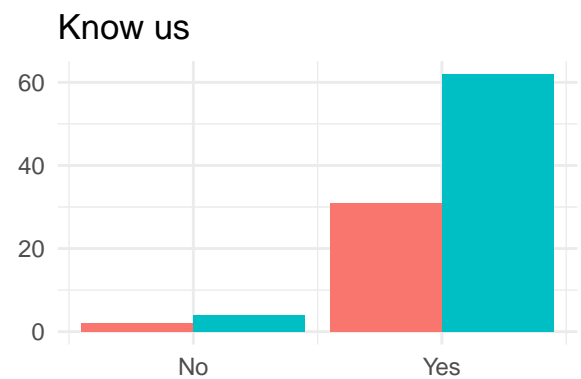
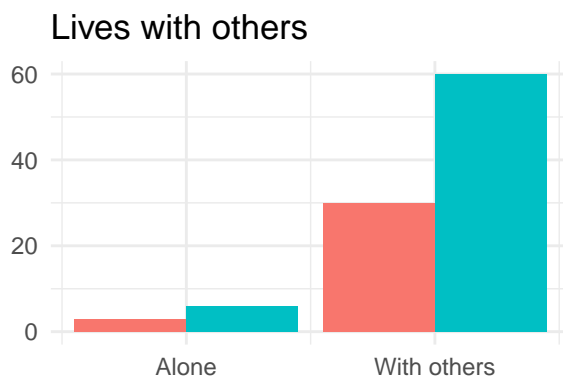
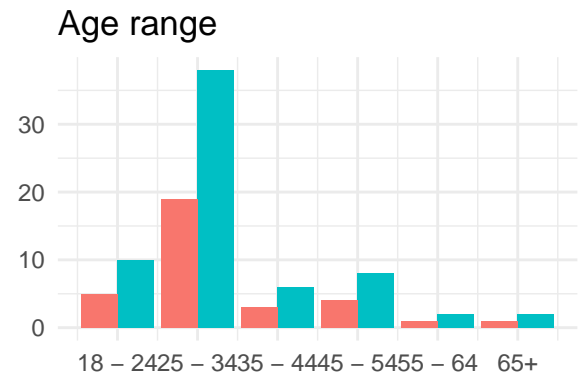
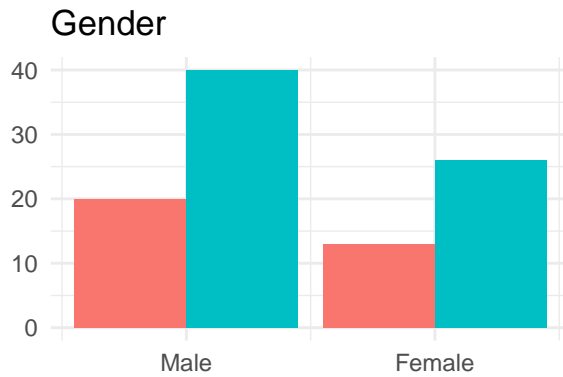
d.age <- d[, c("age_range", "treatment2")]
p_age <- ggplot(d.age, aes(x=age_range, fill = factor(treatment2))) +
  geom_bar(stat="count", position=position_dodge()) +
  theme_minimal() + theme(legend.position="none") +
  xlab("") + ylab("") + ggtitle("Age range") +
  # guides(fill = guide_legend(title = "Assignment")) +
  # scale_fill_discrete(labels = c("Control", "Treatment")) +
  scale_x_continuous(breaks = c(0, 1, 2, 3, 4, 5),
                    labels = c('18 - 24',
                              "25 - 34",
                              "35 - 44",
                              "45 - 54",
                              "55 - 64",
                              "65+"))

d.others <- d[, c("lives_with_others", "treatment2")]
p_others <- ggplot(d.others, aes(x=lives_with_others, fill = factor(treatment2))) +
  geom_bar(stat="count", position=position_dodge()) +
  theme_minimal() + theme(legend.position="none") +
  xlab("") + ylab("") + ggtitle("Lives with others") +
  # guides(fill = guide_legend(title = "Assignment")) +
  # scale_fill_discrete(labels = c("Control", "Treatment")) +
  scale_x_continuous(breaks = c(0, 1),
                    labels = c('Alone', 'With others'))

d.know_us <- d[, c("know_us", "treatment2")]
p_know_us <- ggplot(d.know_us, aes(x=know_us, fill = factor(treatment2))) +
  geom_bar(stat="count", position=position_dodge()) +
  theme_minimal() + theme(legend.position="none") +
  xlab("") + ylab("") + ggtitle("Know us") +
  # guides(fill = guide_legend(title = "Assignment")) +
  # scale_fill_discrete(labels = c("Control", "Treatment")) +
  scale_x_continuous(breaks = c(0, 1),
                    labels = c('No', 'Yes'))

# p_gender
grid.arrange(p_gender_no_legend, p_age, p_others, p_know_us,
             ncol = 2)

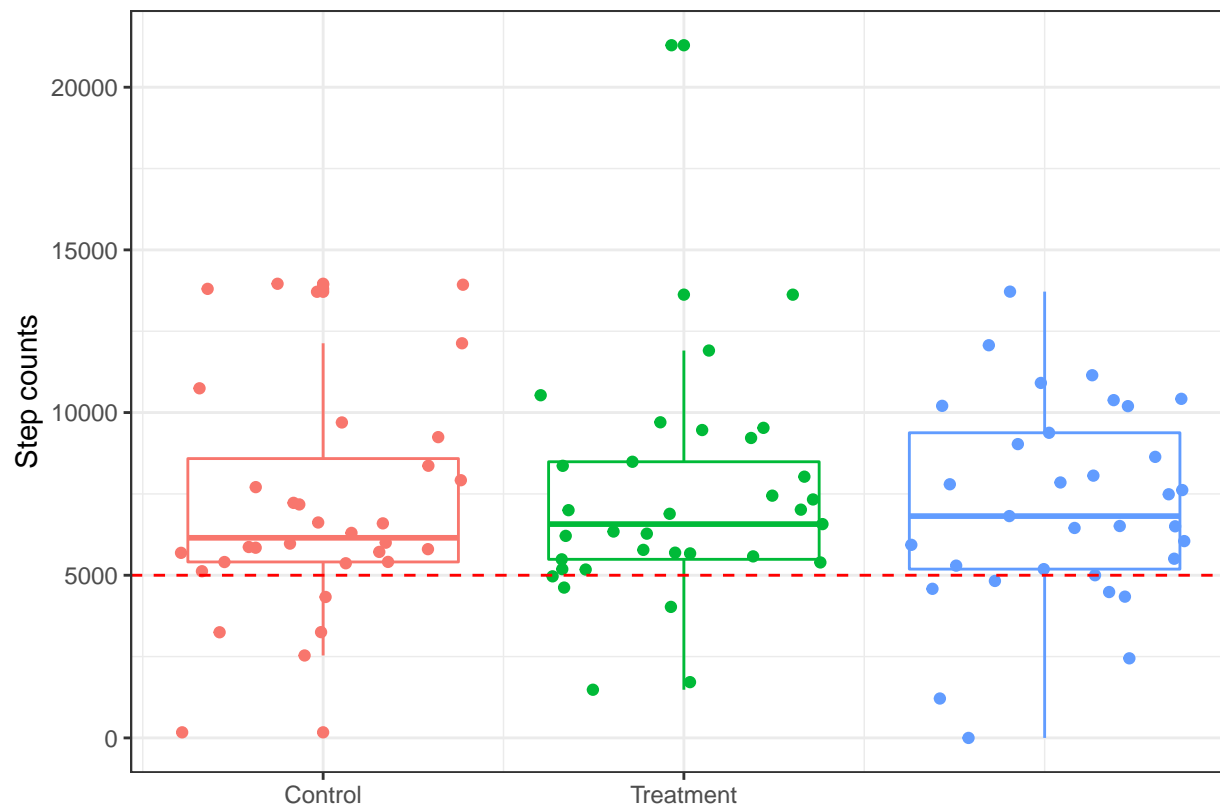
```



```
# control and digital and in person distribution
ggplot(d, aes(x=treatment, y=steps, colour = factor(treatment))) +
  geom_boxplot() + geom_jitter() +
  geom_hline(yintercept=5000, linetype="dashed", color = "red") +
  xlab("") + ylab("Step counts") + theme_bw() +
    scale_x_continuous(breaks = c(0, 1),
                      labels = c('Control', 'Treatment')) +
  theme(legend.position="none")
```

```
## Warning: Removed 1 rows containing non-finite values (stat_boxplot).
```

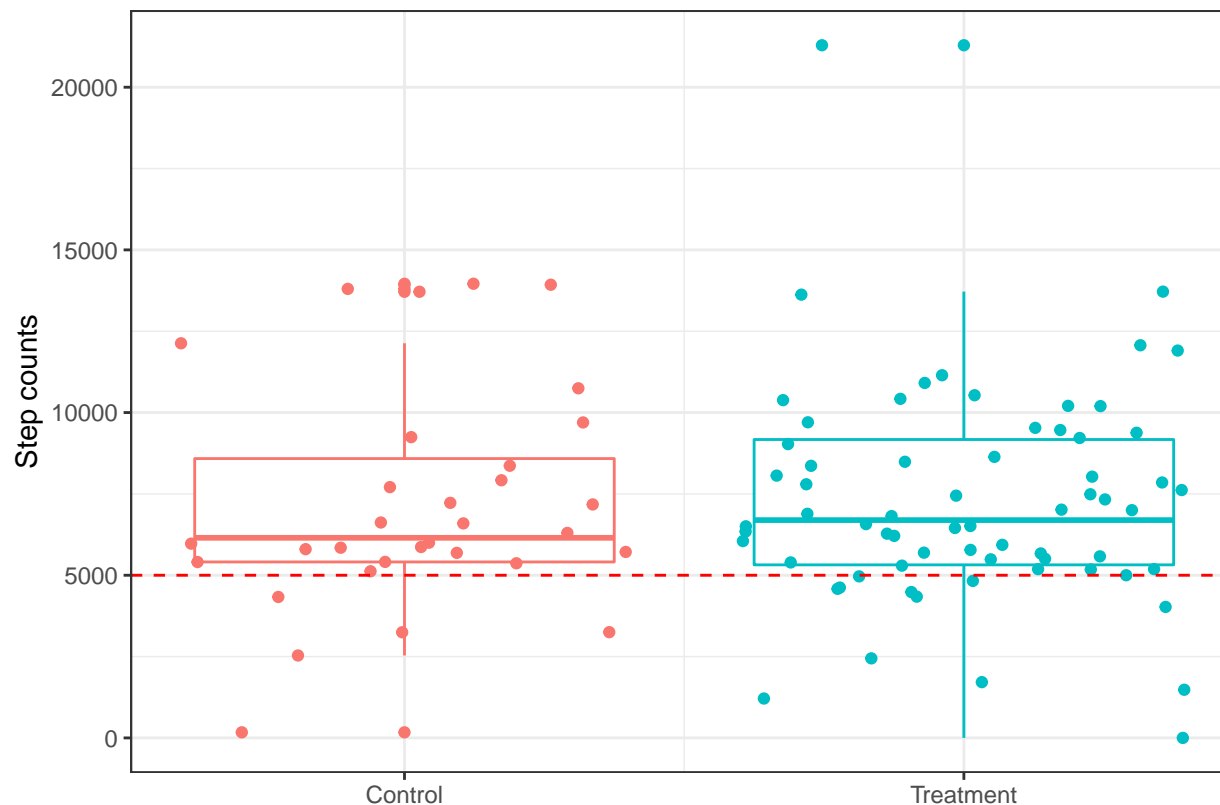
```
## Warning: Removed 1 rows containing missing values (geom_point).
```



```
# control and treatment (digital+in person) when time component removed
ggplot(d, aes(x=treatment2, y=steps, colour = factor(treatment2))) +
  geom_boxplot() + geom_jitter() +
  geom_hline(yintercept=5000, linetype="dashed", color = "red") +
  xlab("") + ylab("Step counts") + theme_bw() +
  scale_x_continuous(breaks = c(0, 1),
                     labels = c('Control', 'Treatment')) +
  theme(legend.position="none")
```

```
## Warning: Removed 1 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



For control vs digital and control vs in person

```
# d$treatment <- factor(d$treatment)
d$userId <- factor(d$userId)
fit_3 <- lm(outcome ~ treatment + userId , d)
# robust se
se_3 <- sqrt(diag(vcovHC(fit_3, type = 'HC'))))

fit_3_covariates <- lm(outcome ~ treatment + age_range + gender + lives_with_others + know_us + location)
# robust se
se_3_covariates <- sqrt(diag(vcovHC(fit_3_covariates, type = 'HC'))))

# ATE (standard error)
print(paste0("Estimated effect of treatment (control, in person, digital): ", signif(fit_3$coefficients
" (", signif(se_3[2], 3), ")"))
```

```
## [1] "Estimated effect of treatment (control, in person, digital): -0.0465 (0.0379)"
```

```
print(paste0("Estimated effect of treatment (control, in person, digital) + covariates: ", signif(fit_3
" (", signif(se_3_covariates[2], 3), ")"))
```

```
## [1] "Estimated effect of treatment (control, in person, digital) + covariates: 0.0468 (0.0465)"
```

```
stargazer(fit_3,
  se=list(se_3),
  omit = c("treatment0"),
  dep.var.labels=c("Steps > 5000"),
  # covariate.labels=c('Commit digitally', 'Commit in person', "User ID", "Constant"),
  omit.stat=c("all"))
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu
 % Date and time: Thu, Dec 20, 2018 - 21:44:46

```
stargazer(fit_3, fit_3_covariates,
  se=list(se_3, se_3_covariates),
  dep.var.labels=c("Steps > 5000"),
  column.labels = c("User ID", "Covariates"),
  # covariate.labels=c('Commit digitally', 'Commit in person', "User ID", "Age range", "Gender"),
  omit.stat=c("all"))
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu
 % Date and time: Thu, Dec 20, 2018 - 21:44:46

test hypothesis that telling others make it more likely to take >5000 steps (control vs treatment)

```
#suppress intercept term
fit_2 <- lm(outcome ~ treatment2 + userId, d)
# robust se
se_2 <- sqrt(diag(vcovHC(fit_2, type = 'HC'))))

fit_2_covariates <- lm(outcome ~ treatment2 + age_range + gender + lives_with_others + know_us + location, d)
# robust se
se_2_covariates <- sqrt(diag(vcovHC(fit_2_covariates, type = 'HC'))))

# ATE (standard error)
print(paste0("Estimated effect of treatment (control, treatment): ", signif(fit_2$coefficients[2], 3),
  " (", signif(se_2[2], 3), ")"))
```

```
## [1] "Estimated effect of treatment (control, treatment): -0.0469 (0.0642)"
```

```
print(paste0("Estimated effect of treatment (control, treatment) + covariates: ", signif(fit_2_covariates$coefficients[2], 3),
  " (", signif(se_2_covariates[2], 3), ")"))
```

```
## [1] "Estimated effect of treatment (control, treatment) + covariates: -0.0447 (0.073)"
```

```
stargazer(fit_2,
  se=list(se_2),
  dep.var.labels=c("Steps > 5000"),
  # covariate.labels=c("Social commitment", "User ID"),
  omit.stat=c("all"))
```

Table 2:

	<i>Dependent variable:</i>
	Steps > 5000
treatment	-0.047 (0.038)
userId2	0.333 (0.254)
userId3	0.667*** (0.254)
userId6	1.000*** (0.031)
userId13	1.000*** (0.031)
userId14	0.333 (0.254)
userId17	0.333 (0.254)
userId19	1.000*** (0.031)
userId22	0.667*** (0.254)
userId25	1.000*** (0.031)
userId26	1.000*** (0.031)
userId28	1.000*** (0.031)
userId33	1.000*** (0.031)
userId39	1.000*** (0.031)
userId45	1.000*** (0.031)
userId47	0.333 (0.292)
userId54	1.000*** (0.031)
userId56	1.000*** (0.031)
userId57	1.000***

Table 3:

	<i>Dependent variable:</i>	
	Steps > 5000	
	User ID	Covariates
	(1)	(2)
treatment	−0.047 (0.038)	−0.045 (0.046)
userId2	0.333 (0.254)	
userId3	0.667*** (0.254)	
userId6	1.000*** (0.031)	
userId13	1.000*** (0.031)	
userId14	0.333 (0.254)	
userId17	0.333 (0.254)	
userId19	1.000*** (0.031)	
userId22	0.667*** (0.254)	
userId25	1.000*** (0.031)	
userId26	1.000*** (0.031)	
userId28	1.000*** (0.031)	
userId33	1.000*** (0.031)	
userId39	1.000*** (0.031)	
userId45	1.000*** (0.031)	
userId47	0.333 (0.292)	
userId54	1.000*** (0.031)	
userId56	1.000*** (0.031)	

Table 4:

	<i>Dependent variable:</i>
	Steps > 5000
treatment2	-0.047 (0.064)
userId2	0.333 (0.260)
userId3	0.667** (0.266)
userId6	1.000*** (0.018)
userId13	1.000*** (0.018)
userId14	0.333 (0.260)
userId17	0.333 (0.260)
userId19	1.000*** (0.018)
userId22	0.667** (0.266)
userId25	1.000*** (0.018)
userId26	1.000*** (0.018)
userId28	1.000*** (0.018)
userId33	1.000*** (0.018)
userId39	1.000*** (0.018)
userId45	1.000*** (0.018)
userId47	0.333 (0.279)
userId54	1.000*** (0.018)
userId56	1.000*** (0.018)
userId57	1.000***

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu
% Date and time: Thu, Dec 20, 2018 - 21:44:46

```
stargazer(fit_2, fit_2_covariates,
  se=list(se_2, se_2_covariates),
  dep.var.labels=c("Steps > 5000"),
  column.labels = c("User ID", "Covariates"),
  # covariate.labels=c("Treatment", "User ID", "Age range", "Gender", "Has housemate", "Knows u
  omit.stat=c("all"))
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu
% Date and time: Thu, Dec 20, 2018 - 21:44:46

power calculations

```
### Control vs digital
# since we fail to reject the null hypothesis,
# let's calculate number of subjects needed for 80% power
effect_size_digital <- cohensD(d[treatment == 0]$steps, d[treatment == 1]$steps)
#power we got from our experiment
pwr.t2n.test(n1 = nrow(d[treatment == 0,]), n2 = nrow(d[treatment == 1,]), d = effect_size_digital, sig
```

```
##
##      t test power calculation
##
##      n1 = 33
##      n2 = 33
##      d = 0.03394626
##      sig.level = 0.05
##      power = 0.05211626
##      alternative = two.sided
```

```
# 80% powered test
pwr.t.test(power = 0.8, d = effect_size_digital, sig.level = 0.05, type = "two.sample")
```

```
##
##      Two-sample t test power calculation
##
##      n = 13623.33
##      d = 0.03394626
##      sig.level = 0.05
##      power = 0.8
##      alternative = two.sided
##
## NOTE: n is number in *each* group
```

```
#
#
#
#
```

Table 5:

	<i>Dependent variable:</i>	
	Steps > 5000	
	User ID	Covariates
	(1)	(2)
treatment2	−0.047 (0.064)	−0.045 (0.073)
userId2	0.333 (0.260)	
userId3	0.667** (0.266)	
userId6	1.000*** (0.018)	
userId13	1.000*** (0.018)	
userId14	0.333 (0.260)	
userId17	0.333 (0.260)	
userId19	1.000*** (0.018)	
userId22	0.667** (0.266)	
userId25	1.000*** (0.018)	
userId26	1.000*** (0.018)	
userId28	1.000*** (0.018)	
userId33	1.000*** (0.018)	
userId39	1.000*** (0.018)	
userId45	1.000*** (0.018)	
userId47	0.333 (0.279)	
userId54	1.000*** (0.018)	
userId56	1.000*** (0.018)	

```

### Control vs in person
# since we fail to reject the null hypothesis,
# let's calculate number of subjects needed for 80% power
effect_size_person <- cohensD(d[treatment == 0]$steps, d[treatment == 2]$steps)
#power we got from our experiment
pwr.t2n.test(n1 = nrow(d[treatment == 0,]), n2 = nrow(d[treatment == 2,]), d = effect_size_person, sig.

##
##      t test power calculation
##
##          n1 = 33
##          n2 = 33
##          d = 0.01871318
##      sig.level = 0.05
##          power = 0.05064253
##      alternative = two.sided

# 80% powered test
pwr.t.test(power = 0.8, d = effect_size_person, sig.level = 0.05, type = "two.sample")

##
##      Two-sample t test power calculation
##
##          n = 44828.14
##          d = 0.01871318
##      sig.level = 0.05
##          power = 0.8
##      alternative = two.sided
##
## NOTE: n is number in *each* group

### extra plots
# day1
pd1 <- ggplot(df, aes(x=day1_treatment, y=day1_steps, colour = factor(day1_treatment))) +
  geom_boxplot() + geom_jitter() +
  geom_hline(yintercept=5000, linetype="dashed", color = "red") +
  xlab("") + ylab("Step counts") + theme_bw() +
  scale_x_continuous(breaks = c(0, 1, 2),
    labels = c(0, 1, 2)) +
  # labels = c('Control', 'In person', 'Through digital means')) +
  theme(legend.position="none") + ggtitle("Step count - day 1")
# day2
pd2 <- ggplot(df, aes(x=day2_treatment, y=day2_steps, colour = factor(day2_treatment))) +
  geom_boxplot() + geom_jitter() +
  geom_hline(yintercept=5000, linetype="dashed", color = "red") +
  xlab("") + ylab("Step counts") + theme_bw() +
  scale_x_continuous(breaks = c(0, 1, 2),
    labels = c(0, 1, 2)) +
  # labels = c('Control', 'In person', 'Through digital means')) +
  theme(legend.position="none") + ggtitle("Step count - day 2")
# day3
pd3 <- ggplot(df, aes(x=day3_treatment, y=day3_steps, colour = factor(day3_treatment))) +

```

```

geom_boxplot() + geom_jitter() +
geom_hline(yintercept=5000, linetype="dashed", color = "red") +
xlab("") + ylab("Step counts") + theme_bw() +
  scale_x_continuous(breaks = c(0, 1, 2),
    labels = c(0, 1, 2)) +
  #           labels = c('Control', 'In person', 'Through digital means')) +
theme(legend.position="none") + ggtitle("Step count - day 3")

```