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"
d2$`Living Situation`[is.na(d2$`Living Situation`)] <- "Other"
d2$Age[is.na(d2$Age)] <- "Other"
d2$Q17[is.na(d2$Q17)] <- "Other"
# stringsAsFactors = F)
names(d2) <- str_replace_all(names(d2), c(" " = "." , "," = "" ))
# subset d2 for those who responded (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 = factor(Treatment.Seq),
             day1_treatment = factor(Q6, levels = c('Not applicable', 'Through digital means', 'In perso
                                                    labels = c(0, 1, 2)),
             day2_treatment = factor(Q10, levels = c('Not applicable', 'Through digital means', 'In pers
                                                    labels = c(0, 1, 2)),
             day3_treatment = factor(Q14, levels = c('Not applicable', 'Through digital means', 'In pers
                                                    labels = c(0, 1, 2)),
             day1_steps = as.numeric(gsub("\\,", "", Q7)),
             day2_steps = as.numeric(gsub("\\,", "", Q11)),
             day3_steps = as.numeric(gsub("\\,", "", Q15)),
             age_range = factor(Age, levels = c('18 - 24',
                                                "25 - 34",
                                                "35 - 44",
                                                "45 - 54",
                                                "55 - 64",
                                                "65+", "Other"),
                                labels = c(0, 1, 2, 3, 4, 5, 6)),
             # gender = factor(Gender),
             gender = factor(Gender, levels = c('Male', 'Female', 'Gender non-conforming'),
                                                    labels = c(0, 1, 2)),
             lives_with_others = factor(Living.Situation, levels = c('Alone', 'With others', "Other"),
                                                            labels = c(0, 1, 2)),
             # know_us = factor(Q17),
             know_us = factor(Q17, levels = c('No', 'Yes', "Other"),
                                                     labels = c(0, 1, 2)),
             location_lat = as.double(LocationLatitude),
             location_long = as.double(LocationLongitude),
             submitted_data = Submitted.Data
)]
```

```
## Warning in eval(jsub, SDenv, 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
```

```
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:     89             4          <NA>          <NA>          <NA>
## 4:     69             3             1             0             2
## 5:     85             3             1             0             2
##    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:         NA         NA         NA         1      0                 1
```

```
## 4:       6343       3247      10198         1      0                 1
## 5:      13624       5406       7851         1      1                 1
##    know_us location_lat location_long submitted_data
## 1:       1     41.89250      -87.7895              1
## 2:       1     37.75101      -97.8220              1
## 3:       1     37.97240     -122.3369              0
## 4:       1     40.37070      -74.0084              1
## 5:       1     42.41730      -71.1087              1
```

```
#Covariate Balance Check 1
bal.tab(as.numeric(treatment_seq) ~ gender + age_range + lives_with_others + know_us + location_lat + lo
        data = d2)
```

```
## Balance Measures
##                         Type Corr.Un
## gender_0              Binary  0.0420
## gender_1              Binary -0.0182
## gender_2              Binary -0.1035
## age_range_0           Binary  0.0345
## age_range_1           Binary -0.0282
## age_range_2           Binary  0.0465
## age_range_3           Binary -0.0404
## age_range_4           Binary  0.0327
## age_range_5           Binary -0.1473
## age_range_6           Binary  0.1688
## lives_with_others_0   Binary  0.0253
## lives_with_others_1   Binary -0.0365
## lives_with_others_2   Binary  0.0327
## know_us_0             Binary  0.0588
## know_us_1             Binary -0.1192
## know_us_2             Binary  0.0945
## location_lat         Contin.  0.0157
## location_long        Contin. -0.0480
##
## Sample sizes
##     Total
## All    75
```

```
cov_check <- lm(as.numeric(treatment_seq) ~ gender + age_range + lives_with_others + know_us + location_
               data = d2)
summary(cov_check)
```

```
##
## Call:
## lm(formula = as.numeric(treatment_seq) ~ gender + age_range +
##     lives_with_others + know_us + location_lat + location_long,
##     data = d2)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -3.002 -1.277  0.000  1.455  2.576
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         0.746226   4.806096   0.155    0.877
## gender1             0.005063   0.460705   0.011    0.991
```

```
## gender2            -1.697377   2.019568  -0.840    0.404
## age_range1         -0.157079   0.608945  -0.258    0.797
## age_range2         -0.037232   0.894025  -0.042    0.967
## age_range3          0.019278   0.893476   0.022    0.983
## age_range4          0.568742   1.949203   0.292    0.771
## age_range5         -1.491405   1.416542  -1.053    0.297
## age_range6          2.505524   1.973739   1.269    0.209
## lives_with_others1 -0.316032   0.829493  -0.381    0.705
## lives_with_others2  0.240428   2.063283   0.117    0.908
## know_us1            0.021548   0.683657   0.032    0.975
## know_us2            0.383130   0.861277   0.445    0.658
## location_lat        0.061884   0.093322   0.663    0.510
## location_long      -0.007443   0.015649  -0.476    0.636
##
## Residual standard error: 1.832 on 60 degrees of freedom
## Multiple R-squared:  0.07962,    Adjusted R-squared:  -0.1351
## F-statistic: 0.3707 on 14 and 60 DF,  p-value: 0.9783
```

attrition check

```
lm_attrit <- lm(submitted_data ~ treatment_seq + age_range + gender + lives_with_others + know_us + loca
summary(lm_attrit)
```

```
##
## Call:
## lm(formula = submitted_data ~ treatment_seq + age_range + gender +
##     lives_with_others + know_us + location_lat + location_long,
##     data = d2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.00766 -0.08803  0.02795  0.20232  0.66351
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -1.603817   0.994669  -1.612 0.112596
## treatment_seq2     -0.046842   0.155938  -0.300 0.765014
## treatment_seq3     -0.013064   0.149640  -0.087 0.930750
## treatment_seq4     -0.053922   0.149343  -0.361 0.719436
## treatment_seq5     -0.239999   0.149378  -1.607 0.113855
## treatment_seq6     -0.239732   0.142557  -1.682 0.098306 .
## age_range1          0.153469   0.116962   1.312 0.194928
## age_range2          0.168041   0.170424   0.986 0.328441
## age_range3          0.405011   0.176223   2.298 0.025371 *
## age_range4          0.327384   0.372228   0.880 0.382943
## age_range5          0.265268   0.280916   0.944 0.349148
## age_range6         -0.277031   0.391906  -0.707 0.482625
## gender1            -0.022546   0.087380  -0.258 0.797350
## gender2            -0.243760   0.399468  -0.610 0.544234
## lives_with_others1  0.153391   0.157930   0.971 0.335670
## lives_with_others2  0.472487   0.395141   1.196 0.236927
## know_us1            0.277886   0.134986   2.059 0.044279 *
## know_us2           -0.615430   0.162941  -3.777 0.000392 ***
```

```
## location_lat        0.041077   0.018501   2.220 0.030541 *
## location_long      -0.004258   0.003073  -1.386 0.171432
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3418 on 55 degrees of freedom
## Multiple R-squared:  0.6063, Adjusted R-squared:  0.4703
## F-statistic: 4.458 on 19 and 55 DF,  p-value: 7.006e-06
```

```
# know_us is highly predictive of whether or not people attrited. This makes sense.
```

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))

d2 <- d2[submitted_data == 1]

# 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 submitted_data
## 1:       1     41.89250      -87.7895              1
## 2:       1     37.75101      -97.8220              1
## 3:       1     40.37070      -74.0084              1
## 4:       1     42.41730      -71.1087              1
## 5:       1     42.35760      -71.0514              1
```

```
# n = 30
d_followed_treatment_sequence <- rbindlist(list(subset(df1, treatment_seq == 1 & df1$day1_treatment == a
                                                       & df1$day2_treatment == assigned_treatment_seq[1
                                                       & df1$day3_treatment == assigned_treatment_seq[1
                                                subset(df1, treatment_seq == 2 & df1$day1_treatment == a
                                                       & df1$day2_treatment == assigned_treatment_seq[2
```

```
                                                    & df1$day3_treatment == assigned_treatment_seq[2
                                                subset(df1, treatment_seq == 3 & df1$day1_treatment == a
                                                    & df1$day2_treatment == assigned_treatment_seq[3
                                                    & df1$day3_treatment == assigned_treatment_seq[3
                                                subset(df1, treatment_seq == 4 & df1$day1_treatment == a
                                                    & df1$day2_treatment == assigned_treatment_seq[4
                                                    & df1$day3_treatment == assigned_treatment_seq[4
                                                subset(df1, treatment_seq == 5 & df1$day1_treatment == a
                                                    & df1$day2_treatment == assigned_treatment_seq[5
                                                    & df1$day3_treatment == assigned_treatment_seq[5
                                                subset(df1, treatment_seq == 6 & df1$day1_treatment == a
                                                    & df1$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$day
                                d_not_followed_treatment_sequence$day1_treatment != d_not_followed_tre
                                d_not_followed_treatment_sequence$day2_treatment != d_not_followed_tre

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 submitted_data
## 1:       1      48.2804       11.5768              1
## 2:       1      42.3576      -71.0514              1
## 3:       1      42.3576      -71.0514              1
```

```
d_not_followed_no_NA <- subset(d_not_followed_treatment_sequence, userId %!in% d_not_followed_but_ok$use
# 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
## -8011.1 -2215.5  -140.7  1981.6  6162.7
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)         9223       2210   4.173  0.00028 ***
## day1_treatment1    -1212       2011  -0.603  0.55174
## day1_treatment2    -1422       1694  -0.839  0.40862
## day2_treatment1    -1795       2301  -0.780  0.44220
## day2_treatment2    -2044       1694  -1.207  0.23799
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3406 on 27 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.06684,    Adjusted R-squared:  -0.0714
## F-statistic: 0.4835 on 4 and 27 DF,  p-value: 0.7476
```

```
# 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: -1210 (2010)"
```

```
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: -1420 (1690)"
```

```
# include days1,2 steps as covariates to understand
# subjects' step counts have as a function of
# treatment against waht 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
## -8433.3 -1006.2   120.8  1148.3  5072.7
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     2967.7308  2745.2966   1.081   0.2900
## day1_treatment1 -592.3631  1756.6233  -0.337   0.7388
## day1_treatment2 -784.1472  1480.5725  -0.530   0.6010
## day2_treatment1 -703.6553  2026.6544  -0.347   0.7313
## day2_treatment2 -134.8561  1571.7867  -0.086   0.9323
## day1_steps         0.3476     0.1611   2.158   0.0408 *
## day2_steps         0.2882     0.2121   1.359   0.1863
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 2944 on 25 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.3545, Adjusted R-squared:  0.1996
## F-statistic: 2.288 on 6 and 25 DF,  p-value: 0.06743
```

```
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: -592 (1760)"
```

```
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: -784 (1480)"
```

```
# 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
## -7337.2 -2223.0  -254.6  1440.9  8568.5
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       7565.3     1065.7   7.099 1.05e-08 ***
## day1_treatment1   -228.2     1243.9  -0.183    0.855
## day1_treatment2  -1104.7     1560.4  -0.708    0.483
## day2_treatment1   -272.7     1452.1  -0.188    0.852
## day2_treatment2   -778.9     1242.1  -0.627    0.534
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3519 on 42 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.01946,    Adjusted R-squared:  -0.07393
## F-statistic: 0.2084 on 4 and 42 DF,  p-value: 0.9324
```

```
# 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: -228 (1240)"
```

```
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: -1100 (1560)"
```

```
# include days1,2 steps as covariates to understand
# subjects' step counts have as a function of
# treatment against waht 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
## -8371.9 -1147.5    -9.2  1643.8  4932.4
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     2681.3421  1180.5601   2.271  0.02859 *
## day1_treatment1 -495.0087   948.4029  -0.522  0.60459
## day1_treatment2 -841.6432  1188.4174  -0.708  0.48292
## day2_treatment1 -857.1243  1110.6529  -0.772  0.44481
## day2_treatment2 -560.6809   951.8148  -0.589  0.55913
## day1_steps         0.2537     0.1268   2.001  0.05223 .
## day2_steps         0.4527     0.1368   3.308  0.00199 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2675 on 40 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.4601, Adjusted R-squared:  0.3792
## F-statistic: 5.682 on 6 and 40 DF,  p-value: 0.0002417
```

```
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: -495 (948)"
```

```
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: -842 (1190)"
```

We do not see that the previous days' treatment assignments to predict the last day's step count is highgly predicitive 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: Fri, Dec 21, 2018 - 14:15:07

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"
```

Table 1:

	Dependent variable:	
	Steps - Day 3	
	(1)	(2)
Treatment - Day 1	−228.177	−495.009
	(1,243.893)	(948.403)
Treatment - Day 2	−1,104.705	−841.643
	(1,560.442)	(1,188.417)
Steps - Day 1	−272.684	−857.124
	(1,452.143)	(1,110.653)
Steps - Day 2	−778.868	−560.681
	(1,242.131)	(951.815)
day1_steps		0.254*
		(0.127)
day2_steps		0.453***
		(0.137)
Constant	7,565.343***	2,681.342**
	(1,065.706)	(1,180.560)
Note:	*p<0.1; **p<0.05; ***p<0.01	

```
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)
```

```
##    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 submitted_data treatment2 outcome
## 1:       1     36.05251      -79.1077              1          0       1
## 2:       1     42.35760      -71.0514              1          0       1
## 3:       1     42.37700      -71.1256              1          0       1
## 4:       1     42.35760      -71.0514              1          0       1
## 5:       1     42.61240      -83.0345              1          0       1
```

```
#Covariate Balance Check
#bal.tab(treatment_seq ~ gender + age_range + lives_with_others + know_us + location_lat + location_lon
#        data = d)
#cov_check <- lm(treatment_seq ~ gender + age_range + lives_with_others + know_us + location_lat + loca
#                data = d)
#summary(cov_check)
```

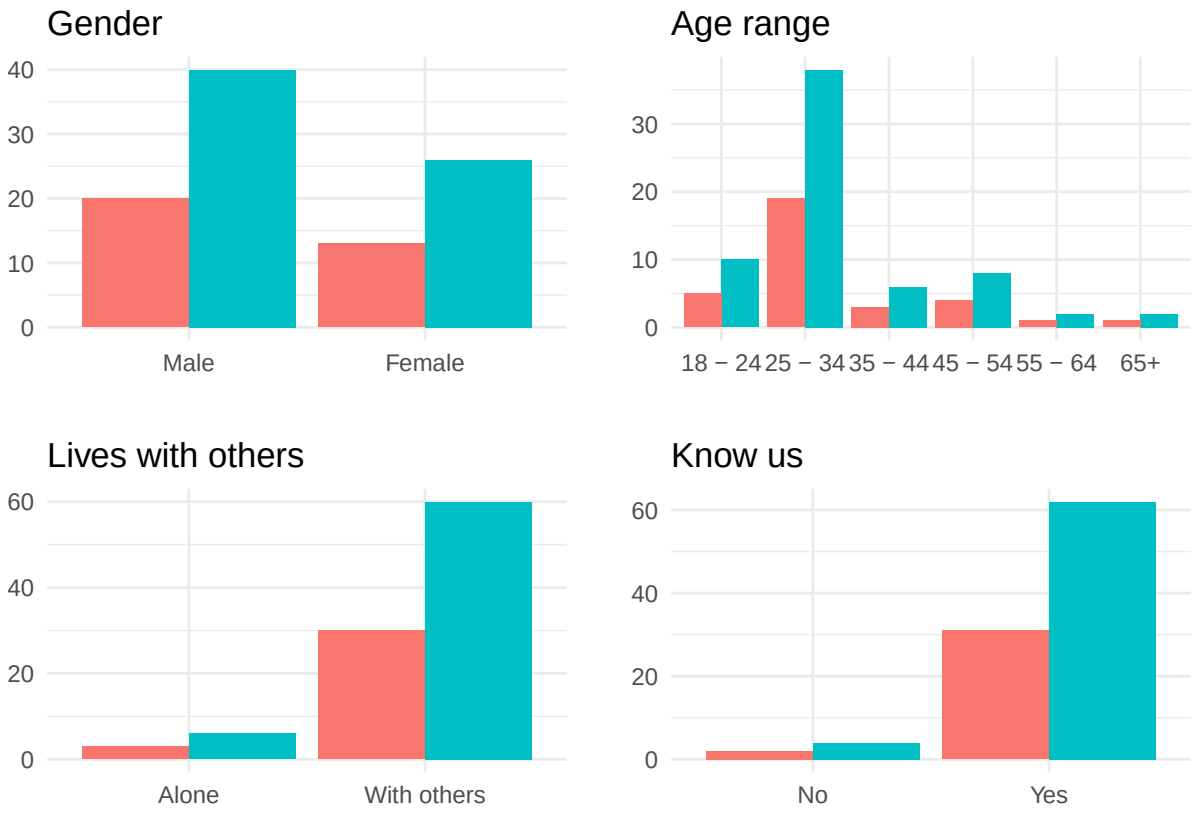
Make some pretty plots to show distribution, populatin etc.

```
# 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_discrete(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") +
```

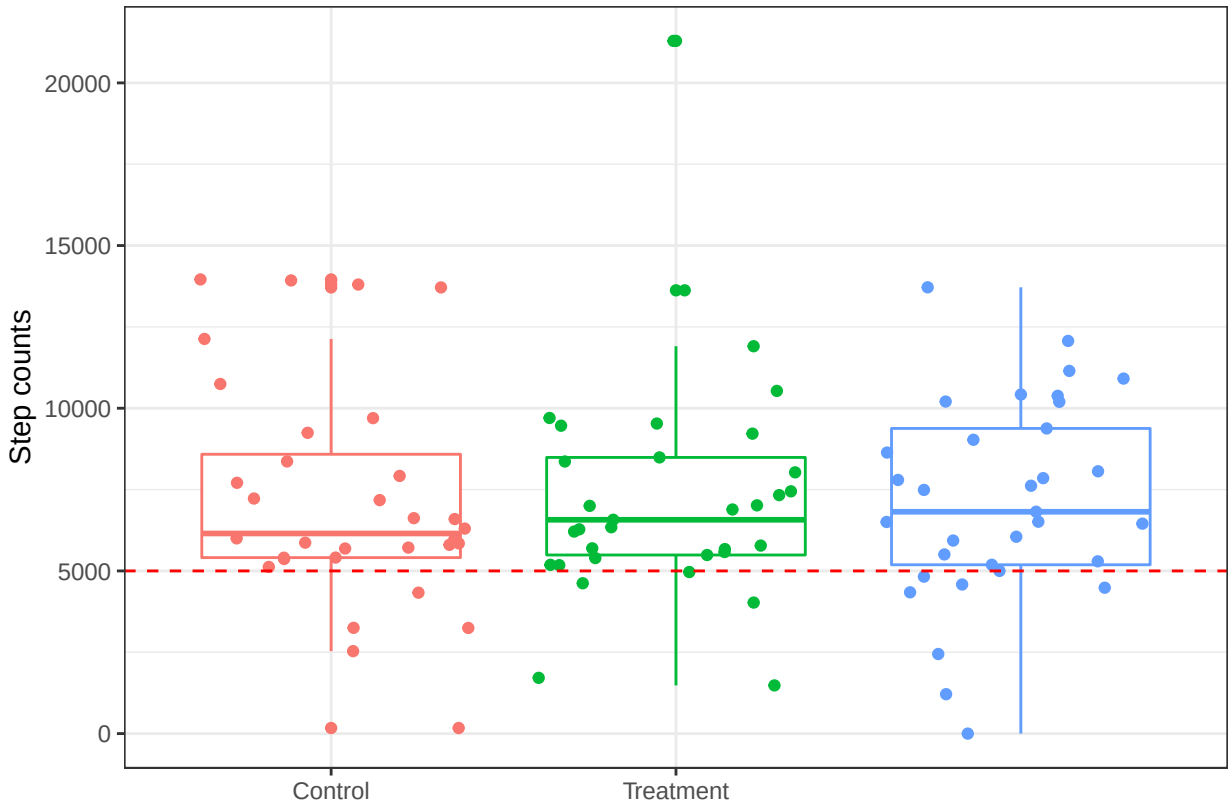
```
  # guides(fill = guide_legend(title = "Assignment")) +
  # scale_fill_discrete(labels = c("Control", "Treatment")) +
  scale_x_discrete(breaks = c(0, 1, 2),
                   labels = c('Male', 'Female', 'Gender\n non-conforming'))


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_discrete(breaks = c(0, 1, 2, 3, 4, 5, 6 ),
                   labels = c('18 - 24',
                              "25 - 34",
                              "35 - 44",
                              "45 - 54",
                              "55 - 64",
                              "65+", "NA"))

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_discrete(breaks = c(0, 1, 2),
                   labels = c('Alone', 'With others', "NA"))

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_discrete(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_discrete(breaks = c(0, 1),
                     labels = c('Control', 'Treatment')) +
  theme(legend.position="none")
```

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

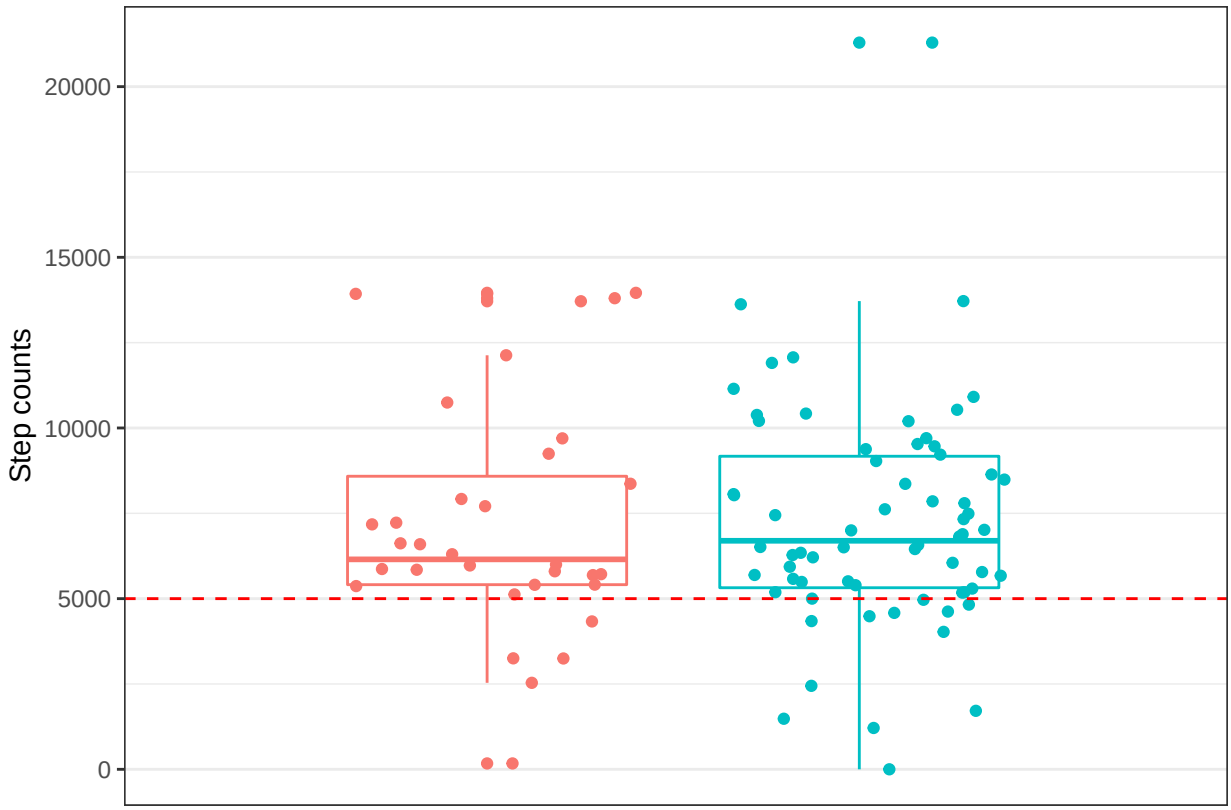
```
## Warning: Removed 1 rows containing missing values (geom_point).
```




```
# 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_discrete(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)
# se clustered based on userID
se_3 <- coeftest(fit_3, vcovHC(fit_3, type = 'HC', cluster = "userID"))

fit_3_covariates <- lm(outcome ~ treatment + age_range + gender + lives_with_others + know_us + locatio
# 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[2,2], 3), ")"))
```

```
## [1] "Estimated effect of treatment (control, in person, digital): -0.00142 (0.0652)"
```

```
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: -0.0884 (0.078)"
```

```
stargazer(fit_3,
          se=list(se_3[,2]),
          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: Fri, Dec 21, 2018 - 14:15:10

```
stargazer(fit_3, fit_3_covariates,
          se=list(se_3[,2], 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: Fri, Dec 21, 2018 - 14:15:10

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)
#  se clustered based on userID
se_2 <- coeftest(fit_2, vcovHC(fit_2, type = 'HC', cluster = "userID"))

fit_2_covariates <- lm(outcome ~ treatment2 + age_range + gender + lives_with_others + know_us + locati
# 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 (-0.0469)"
```

```
print(paste0("Estimated effect of treatment (control, treatment) + covariates: ", signif(fit_2_covariat
" (", signif(se_2_covariates[2], 3), ")"))
```

```
## [1] "Estimated effect of treatment (control, treatment) + covariates: -0.0429 (0.0737)"
```

```
stargazer(fit_2,
          se=list(se_2[,2]),
          dep.var.labels=c("Steps > 5000"),
          # covariate.labels=c("Social commitment", "User ID"),
          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: Fri, Dec 21, 2018 - 14:15:11

```
stargazer(fit_2, fit_2_covariates,
          se=list(se_2[,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 2:

	Dependent variable:
	Steps > 5000
treatment1	−0.001
	(0.065)
treatment2	−0.092
	(0.076)
userId2	0.333
	(0.261)
userId3	0.667***
	(0.248)
userId6	1.000***
	(0.035)
userId13	1.000***
	(0.035)
userId14	0.333
	(0.261)
userId17	0.333
	(0.261)
userId19	1.000***
	(0.035)
userId22	0.667***
	(0.248)
userId25	1.000***
	(0.035)
userId26	1.000***
	(0.035)
userId28	1.000***
	(0.035)
userId33	1.000***
	(0.035)
userId39	1.000***
	(0.035)
userId45	1.000***
	(0.035)
userId47	0.333
	(0.298)
userId54	1.000***
	(0.035)
userId56	1.000***

Table 3:

	Dependent variable:	
	Steps > 5000	
	User ID	Covariates
	(1)	(2)
treatment1	−0.001	0.003
	(0.065)	(0.078)
treatment2	−0.092	−0.088
	(0.076)	(0.092)
userId2	0.333	
	(0.261)	
userId3	0.667***	
	(0.248)	
userId6	1.000***	
	(0.035)	
userId13	1.000***	
	(0.035)	
userId14	0.333	
	(0.261)	
userId17	0.333	
	(0.261)	
userId19	1.000***	
	(0.035)	
userId22	0.667***	
	(0.248)	
userId25	1.000***	
	(0.035)	
userId26	1.000***	
	(0.035)	
userId28	1.000***	
	(0.035)	
userId33	1.000***	
	(0.035)	
userId39	1.000***	
	(0.035)	
userId45	1.000***	
	(0.035)	
userId47	0.333	
	(0.298)	
userId54	1.000***	
	(0.035)	

Table 4:

	Dependent variable:
	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: Fri, Dec 21, 2018 - 14:15:11

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
```

```
#
#
#
#
###  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
```

Table 5:

	Dependent variable:	
	Steps > 5000	
	User ID	Covariates
	(1)	(2)
treatment2	−0.047	−0.043
	(0.064)	(0.074)
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)	

```
##           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")
```