

R Notebook

Libraries

```
# Load libraries  
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --  
## v dplyr      1.1.2      v readr      2.1.4  
## v forcats    1.0.0      v stringr   1.5.0  
## v ggplot2    3.4.2      v tibble    3.2.1  
## v lubridate  1.9.2      v tidyr     1.3.0  
## v purrr      1.0.1  
## -- Conflicts ----- tidyverse_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag()    masks stats::lag()  
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(googleSheets4)  
library(clipr)
```

```
## Welcome to clipr. See ?write_clip for advisories on writing to the clipboard in R.
```

```
library(sandwich)
```

```
## Warning: package 'sandwich' was built under R version 4.3.3
```

```
library(lmtest)
```

```
## Loading required package: zoo  
##  
## Attaching package: 'zoo'  
##  
## The following objects are masked from 'package:base':  
##  
##      as.Date, as.Date.numeric
```

Function for robust standard error

```
library(stargazer)
```

```
##
## Please cite as:

## Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary Statistics Tables.

## R package version 5.2.3. https://CRAN.R-project.org/package=stargazer

fn_robust_errors = function(mod){
  mod$vcovHC_ <- vcovHC(mod)
  coeftest(mod, vcov. = mod$vcovHC_)

  stargazer(
    mod,
    type = 'text',
    se=list(sqrt(diag(mod$vcovHC_))),
    header=F
  )
}

# Print out confidence intervals
fn_confidence <- function(m, se) {
  cat(
    paste0(
      "Mean: ", m, "\n",
      "Low: ", m - (1.96 * se), "\n",
      "High: ", m + (1.96 * se)
    )
  )
}
}
```

Prolific Respondent Pool

Day 1 ETL

```
df_survey_1 = read_sheet('https://docs.google.com/spreadsheets/d/1DnjmX-oINxHNkPx360bpLDDIZUCb9zVBguSw4...')

## ! Using an auto-discovered, cached token.

## To suppress this message, modify your code or options to clearly consent to
## the use of a cached token.

## See gargle's "Non-interactive auth" vignette for more details:
## <https://gargle.r-lib.org/articles/non-interactive-auth.html>

## i The googlesheets4 package is using a cached token for 'bwesche@berkeley.edu'.

## v Reading from "Coffee Drinking Experiment - Prolific Day 1 (Responses)".

## v Range 'Form Responses 1'.
```

```

# New column names
cols_rename = c("timestamp", "prolific_id", "coffee_screener", "wakeup_time", "coffee_time", "hours_wok

# Rename columns
colnames(df_survey_1) = cols_rename

# Processing
df_survey_1 = df_survey_1 %>%
  mutate(
    prolific_id = as.character(prolific_id), # Unnest ID
    # Parse # from the coffee amount
    q_coffee_amount = str_extract(q_coffee_amount, "\\d+"),
    hours_woke_up = as.character(hours_woke_up),
    # Parse # from metric fields
    across(c(hours_woke_up, q_difficult_task, hours_woke_up, q_awake, q_difficult_task, q_tired), ~as.n
  ) %>%
  # Filter out duplicate entries and only take each person's first one from the first day
  arrange(prolific_id, timestamp) %>%
  group_by(prolific_id) %>% mutate(n_surveys = n()) %>% # Total number of surveys taken
  mutate(surv_num = row_number()) %>%
  ungroup() %>% filter(surv_num==1) %>%
  # Filter out blank rows
  filter(!is.na(timestamp)) %>%
  # Create alertness score (avg of 3 metrics)
  mutate(alertness_score = rowMeans(across(c(q_awake, q_difficult_task, q_tired)))) %>%
  # Filter out non-compliers (people who didn't drink coffee within 15-45 minutes of waking)
  mutate(coff_min_wakeup = coffee_time-wakeup_time) %>%
  mutate(coffee_min_wakeup = as.numeric(str_extract(coff_min_wakeup, "^\\d+"))) %>%
  mutate(coffee_min_wakeup = coffee_min_wakeup/60) %>%
  filter(coffee_min_wakeup>=15 & coffee_min_wakeup<=45)

```

Day 2 ETL

```

# Survey day 2
df_survey_2 = read_sheet('https://docs.google.com/spreadsheets/d/1WW7nzk-jSw7TsxIh-9HgEyjXp403PIAKV70Dm

## v Reading from "Coffee Drinking Experiment - Prolific Day 2 (Responses)".

## v Range 'Form Responses 1'.

# New column names
cols_rename = c("timestamp", "prolific_id", "coffee_screener", "wakeup_time", "coffee_time", "hours_wok
# Add "2" suffix to each column to flag as 2nd day responses
cols_rename = paste0(cols_rename, "_2")

# Rename columns
colnames(df_survey_2) = cols_rename

# Processing

```

```

df_survey_2 = df_survey_2 %>%
  mutate(
    prolific_id_2 = as.character(prolific_id_2), # Unnest ID
    q_coffee_amount_2 = str_extract(q_coffee_amount_2, "\\d+"),
    hours_woke_up_2 = as.character(hours_woke_up_2),
    across(c(hours_woke_up_2, q_difficult_task_2, hours_woke_up_2, q_awake_2, q_difficult_task_2, q_tired_2),
           ) %>%
    # Filter out duplicate entries and only take each person's first one
    arrange(prolific_id_2, timestamp_2) %>%
    group_by(prolific_id_2) %>% mutate(n_surveys_2 = n()) %>% # Total number of surveys taken
    mutate(surv_num_2 = row_number()) %>%
    ungroup() %>% filter(surv_num_2==1) %>%
    # Filter out blank rows
    filter(!is.na(timestamp_2)) %>%
    # Create composite alertness score
    mutate(alertness_score_2 = rowMeans(across(c(q_awake_2, q_difficult_task_2, q_tired_2)))) %>%
    # Filter out non-compliers (people who didn't drink coff within 2-5 hours of waking up)
    mutate(coff_min_wakeup = coffee_time_2-wakeup_time_2) %>%
    mutate(coffee_min_wakeup = as.numeric(str_extract(coff_min_wakeup, "^\\d+"))) %>%
    mutate(coffee_min_wakeup = coffee_min_wakeup/3600) %>%
    filter(coffee_min_wakeup>=2 & coffee_min_wakeup<=4)

```

Merge data sources

```

df_survey_merged = df_survey_1 %>%
  inner_join(df_survey_2, by = c("prolific_id" = "prolific_id_2")) %>%
  # Calculate treatment effect for main metrics
  mutate(
    hours_woke_up_delta = hours_woke_up_2 - hours_woke_up,
    q_awake_delta = q_awake_2 - q_awake,
    q_difficult_task_delta = q_difficult_task_2 - q_difficult_task,
    # q_alertness_delta = q_alert
    q_alertness_score_delta = alertness_score_2 - alertness_score
  ) %>%
  filter(coffee_screener=="Yes")

```

T-Test Results

```

# Look at simple average
df_survey_merged %>%
  select(contains("delta")) %>%
  # pivot_longer(is.numeric) %>%
  # ggplot(aes(value)) + geom_histogram() + facet_wrap(~name) + theme_light()
  summarise(across(is.numeric, ~mean(.x, na.rm=T)), .groups = "drop")

```

```

## Warning: There was 1 warning in `summarise()`.
## i In argument: `across(is.numeric, ~mean(.x, na.rm = T))`.
## Caused by warning:
## ! Use of bare predicate functions was deprecated in tidysselect 1.1.0.

```

```
## i Please use wrap predicates in `where()` instead.
## # Was:
## data %>% select(is.numeric)
##
## # Now:
## data %>% select(where(is.numeric))

## # A tibble: 1 x 4
##   hours_woke_up_delta q_awake_delta q_difficult_task_delta
##             <dbl>         <dbl>         <dbl>
## 1             4.98         0.0909         0.114
## # i 1 more variable: q_alertness_score_delta <dbl>

# Use this!!
t.test(df_survey_merged$alertness_score_2, df_survey_merged$alertness_score)

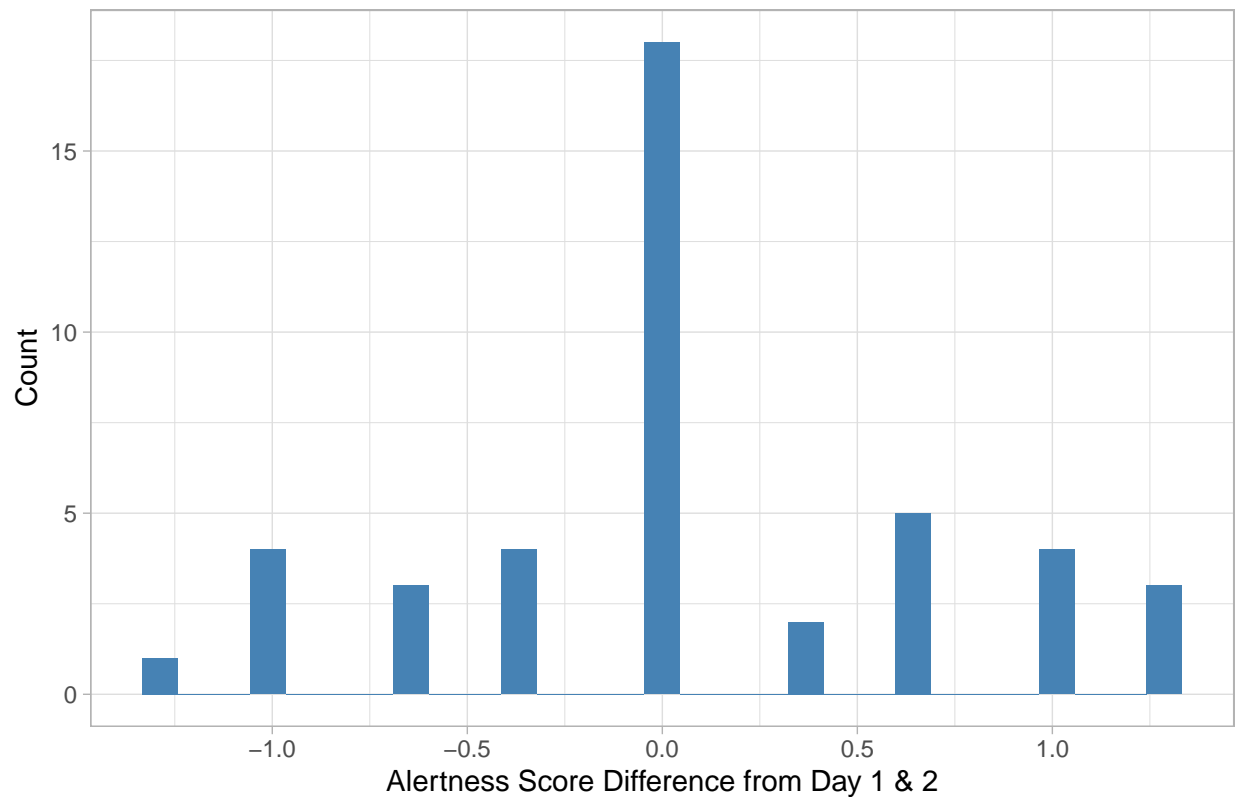
##
## Welch Two Sample t-test
##
## data: df_survey_merged$alertness_score_2 and df_survey_merged$alertness_score
## t = 0.51087, df = 85.815, p-value = 0.6108
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.2190462  0.3705614
## sample estimates:
## mean of x mean of y
## 4.060606  3.984848
```

Histogram of dependent variable

```
df_survey_merged %>%
  ggplot(aes(q_alertness_score_delta)) + geom_histogram(fill = "steelblue") + theme_light() +
  labs(title = "Distribution of alertness score differences", x = "Alertness Score Difference from Day 1")

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Distribution of alertness score differences



Control for coffee amount

Only two people submitted 4+ cups a day

- There's only stat sig with 4+, but only 2 people so don't reco using it

df_survey_merged

```
## # A tibble: 44 x 37
##   timestamp      prolific_id      coffee_screener wakeup_time
##   <dtm>          <chr>          <chr>          <dtm>
## 1 2025-04-10 10:05:15 5652916bfea9c60012c5~ Yes      1899-12-30 05:00:00
## 2 2025-04-10 10:51:49 56eac3b6e6c248000b17~ Yes      1899-12-30 06:00:00
## 3 2025-04-10 10:39:30 5b5a6c5f7ac6d70001aa~ Yes      1899-12-30 09:07:00
## 4 2025-04-10 10:20:35 5d9b6e36adca6b00193f~ Yes      1899-12-30 05:00:00
## 5 2025-04-10 16:15:11 5dd5b42a2091e95798fb~ Yes      1899-12-30 07:00:00
## 6 2025-04-10 10:58:17 5e175705cfe8dc000b55~ Yes      1899-12-30 07:00:00
## 7 2025-04-10 11:11:10 5e3753e214f39d6dbf4c~ Yes      1899-12-30 05:25:00
## 8 2025-04-10 10:48:54 5eeab6c160ae2d1dcb69~ Yes      1899-12-30 05:30:00
## 9 2025-04-10 10:05:25 6025b3681ff1521a3369~ Yes      1899-12-30 08:30:00
## 10 2025-04-10 10:37:50 60fc5a8eb17e706908c6~ Yes      1899-12-30 06:30:00
## # i 34 more rows
## # i 33 more variables: coffee_time <dtm>, hours_woke_up <dbl>,
## #   q_coffee_amount <chr>, q_awake <dbl>, q_difficult_task <dbl>,
```

```
## #   q_tired <dbl>, q_alertness_diff <chr>, q_additional_comments <chr>,
## #   n_surveys <int>, surv_num <int>, alertness_score <dbl>,
## #   coff_min_wakeup.x <drtn>, coffee_min_wakeup.x <dbl>, timestamp_2 <dtm>,
## #   coffee_screener_2 <chr>, wakeup_time_2 <dtm>, coffee_time_2 <dtm>, ...
```

```
mod_amount = lm(q_difficult_task_delta ~ q_coffee_amount, data = df_survey_merged)
summary(mod_amount)
```

```
##
## Call:
## lm(formula = q_difficult_task_delta ~ q_coffee_amount, data = df_survey_merged)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0000 -0.2692  0.0000  0.7308  1.7308
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.2692     0.1554   1.732  0.09089 .
## q_coffee_amount2 -0.2692     0.2518  -1.069  0.29134
## q_coffee_amount3 -0.2692     0.8075  -0.333  0.74056
## q_coffee_amount4 -2.2692     0.8075  -2.810  0.00763 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7924 on 40 degrees of freedom
## Multiple R-squared:  0.1747, Adjusted R-squared:  0.1128
## F-statistic: 2.822 on 3 and 40 DF,  p-value: 0.05093
```

```
fn_robust_errors(mod_amount)
```

```
## Warning in meatHC(x, type = type, omega = omega): HC3 covariances are
## numerically unstable for hat values close to 1 (and undefined if exactly 1) as
## for observation(s) 20, 44
```

```
##
## =====
##              Dependent variable:
##              -----
##              q_difficult_task_delta
##              -----
## q_coffee_amount2              -0.269
##
## q_coffee_amount3              -0.269
##
## q_coffee_amount4              -2.269
##
## Constant                      0.269
##
```

```
##
## -----
## Observations          44
## R2                    0.175
## Adjusted R2           0.113
## Residual Std. Error   0.792 (df = 40)
## F Statistic           2.822* (df = 3; 40)
## =====
## Note:                  *p<0.1; **p<0.05; ***p<0.01
```

```
# Pull count
df_survey_merged %>%
  count(q_coffee_amount)
```

```
## # A tibble: 4 x 2
##   q_coffee_amount     n
##   <chr>          <int>
## 1 1              26
## 2 2              16
## 3 3               1
## 4 4               1
```

Control for wake-up time

No Stat sig

```
mod_wakeup = lm(q_difficult_task_delta ~ wakeup_time, data = df_survey_merged)
summary(mod_wakeup)
```

```
##
## Call:
## lm(formula = q_difficult_task_delta ~ wakeup_time, data = df_survey_merged)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.94711 -0.26123  0.07454  0.44968  1.65611
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.623e+05  5.667e+04   2.865  0.00649 **
## wakeup_time  7.348e-05  2.565e-05   2.865  0.00649 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7786 on 42 degrees of freedom
## Multiple R-squared:  0.1634, Adjusted R-squared:  0.1435
## F-statistic: 8.205 on 1 and 42 DF, p-value: 0.006495
```

Control for coffee time

No stat sig for coffee time


```
mod_coffee_time = lm(q_difficult_task_delta ~ coffee_time, data = df_survey_merged)
summary(mod_coffee_time)
```

```
##
## Call:
## lm(formula = q_difficult_task_delta ~ coffee_time, data = df_survey_merged)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.98493 -0.27095  0.06461  0.46060  1.68505
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.620e+05  5.746e+04   2.82  0.00731 **
## coffee_time  7.334e-05  2.601e-05   2.82  0.00731 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7805 on 42 degrees of freedom
## Multiple R-squared:  0.1592, Adjusted R-squared:  0.1391
## F-statistic:  7.95 on 1 and 42 DF,  p-value: 0.007306
```