```r
library(readxl)
library(lme4)
```

```
## Loading required package: Matrix
```

```r
library(lmerTest)
```

```
##
## Attaching package: 'lmerTest'

## The following object is masked from 'package:lme4':
##
##     lmer

## The following object is masked from 'package:stats':
##
##     step
```

```r
library(car)
```

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

```r
library(MuMIn)
library(afex)
```

```
## ************
## Welcome to afex. For support visit: http://afex.singmann.science/

## - Functions for ANOVAs: aov_car(), aov_ez(), and aov_4()
## - Methods for calculating p-values with mixed(): 'S', 'KR', 'LRT', and 'PB'
## - 'afex_aov' and 'mixed' objects can be passed to emmeans() for follow-up tests
## - NEWS: emmeans() for ANOVA models now uses model = 'multivariate' as default.
## - Get and set global package options with: afex_options()
## - Set orthogonal sum-to-zero contrasts globally: set_sum_contrasts()
## - For example analyses see: browseVignettes("afex")
## ************

##
## Attaching package: 'afex'

## The following object is masked from 'package:lme4':
##
##     lmer
```

```r
data <- read_excel("../Data/PredictingOutcomes_ParticipantDemographics.xlsx", sheet = "Study 1A")

# print(data)
```

create a map like data structure to store the unique participant id with there corresponding gender

```r
map <- data.frame(unique(data$participant_id), data$gender)
colnames(map) <- c("participant_id","gender")
# map
```

```r
data1 <- read_excel("../Data/PredictingOutcomes_ParticipantPredictions.xlsx", sheet = "Study 1A")
# only tke the data for columns participant_id, prediction_recode, prediction_recode
data1 <- data1[,c(2,3,10)]
# print(data1)
```

```r
df <- merge(data1, map, by = "participant_id")
male <- df[df$gender=='0',]
female <- df[df$gender=='1',]
t.test(male$prediction_recode, female$prediction_recode)
```

```
##
##  Welch Two Sample t-test
##
## data:  male$prediction_recode and female$prediction_recode
## t = 3.8576, df = 2045.1, p-value = 0.0001181
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  2.101319 6.447286
## sample estimates:
## mean of x mean of y
##  53.64212  49.36782
```

```r
df_analyst <- df[df$generator=="analyst",]
male <- df_analyst[df_analyst$gender=='0',]
female <- df_analyst[df_analyst$gender=='1',]
t.test(male$prediction_recode, female$prediction_recode)
```

```
##
##  Welch Two Sample t-test
##
## data:  male$prediction_recode and female$prediction_recode
## t = 2.4113, df = 749.27, p-value = 0.01613
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.8476775 8.2732858
## sample estimates:
## mean of x mean of y
##  53.75096  49.19048
```

```r
df_stock <- df[df$generator=="stock",]
male <- df_stock[df_stock$gender=='0',]
female <- df_stock[df_stock$gender=='1',]
t.test(male$prediction_recode, female$prediction_recode)
```

```
##
##  Welch Two Sample t-test
##
```

```
## data:  male$prediction_recode and female$prediction_recode
## t = 1.7288, df = 515.29, p-value = 0.08444
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.4939742  7.7390139
## sample estimates:
## mean of x mean of y
##  55.71627  52.09375
```

```r
df_bingo <- df[df$generator=="bingo",]
male <- df_bingo[df_bingo$gender=='0',]
female <- df_bingo[df_bingo$gender=='1',]
t.test(male$prediction_recode, female$prediction_recode)
```

```
##
##  Welch Two Sample t-test
##
## data:  male$prediction_recode and female$prediction_recode
## t = 2.2476, df = 772.82, p-value = 0.02489
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.514269 7.610526
## sample estimates:
## mean of x mean of y
##  51.53065  47.46825
```