# Testing

2023-07-14

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

```
library(dbplyr)
library(readxl)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.1     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::ident()  masks dbplyr::ident()
## x dplyr::lag()    masks stats::lag()
## x dplyr::sql()    masks dbplyr::sql()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(ggplot2)
```

```
qData <- read.csv("Effect of Positive copy.csv")
pData <- read_csv("Prolific id copy.csv")
```

```
## New names:
## Rows: 169 Columns: 5
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (2): prolificid, Response_Id lgl (3): ...3, ...4, ...5
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...3`
```

```
## * `` -> `...4`
## * `` -> `...5`
```

```r
demographic <- read.csv("Demographic_Data.csv")
```

```r
cmbndData <- merge(pData, qData, by = "Response_Id", all = TRUE)
cmbndData <- merge(cmbndData, demographic, by = "prolificid", all = TRUE)
```

```r
#Cleaning data
cmbndData <- cmbndData[complete.cases(cmbndData$Response_Id), ] #removing rows with missing values
cmbndData <- cmbndData[complete.cases(cmbndData$prolificid), ]#removing rows with missing values
cmbndData <- unique(cmbndData) #removing duplicates
newData <- cmbndData[, c("prolificid","Describe.memories","Describe.memories.1","Q124","Q125","Q126", "Q
newData_2 <- cmbndData[, c("prolificid","Q1.1","Q2.1","Q3.2","Q114","Q5.1","Current.mood","Age","Sex","E
newData <- merge(newData, newData_2, by = "prolificid", all = TRUE)
```

```r
#Renaming & Reorganizing
newData <- newData %>% rename(Happy_Condition = Describe.memories) #renaming columns
newData <- newData %>% rename(Neutral_Condition = Describe.memories.1) #renaming columns
newData$Happy_Condition <- ifelse(nzchar(newData$Happy_Condition), 1, 2) #If column has text then chang
#1 is Happiness and #2 is Neutral
newData <- newData %>% rename(Condition = Happy_Condition)
newData <- subset(newData, select = -Neutral_Condition) #Deleting Neutral Condition column
newData <- newData %>% rename(Q1 = Q124, Q2 = Q125, Q3 = Q126, Q4 = Q127, Q5 = Q128,Q6 = Q129, Q7 = Q130
newData <- newData %>% rename(G.1 = Q1.1, G.2 = Q2.1, G.3 = Q3.2, G.4 = Q114, G.5 = Q5.1)
```

```r
#Recoding Emotion Managament Test Scores
#Each correct answer receives 2 points
#Total score is out of 36
#if correct answer then participant receives a score of 2, or else participant receives 0
recode_score <- function(x, correct_answer) {
  ifelse(grepl(correct_answer, x), "2", "0")
}

columns_to_process <- c("Q1", "Q2", "Q3", "Q4", "Q5","Q6", "Q7", "Q8", "Q9", "Q10", "Q11", "Q12","Q13",


correct_answers <- c("Contact Wai-Hin and arrange to catch up but also make friends with her replacement
"Make sure she keeps in contact through email, phone or letter writing.","Document the lack of resources
"Talk to a doctor about what the normal rates of development are.",
"Find out if there is some reasonable explanation for the shift changes.", "Understand that people chang

#Creating a loop
for (i in seq_along(columns_to_process)) {
  col <- columns_to_process[i]
  correct_answer <- correct_answers[i]
  newData[[col]] <- recode_score(newData[[col]], correct_answer)
}
```

```r
#Creating new column with aggregate score on Emotion Management Test
EM_scores <- newData %>% select(prolificid, Condition, c(Q1:Q18))
for (col in 3:20) {
  EM_scores[[col]] <- as.numeric(EM_scores[[col]])
}
EM_scores <- EM_scores %>%
  rowwise() %>%
```

```r
  mutate(Total_score = sum(c_across(3:20), na.rm = TRUE)) #Calculating the total score for each partici

#Calculating average score of each group
EM_scores <- EM_scores %>% select(Condition, Total_score, prolificid)
EM_avg <- EM_scores %>% group_by(Condition)
EM_avg_wide <- EM_avg %>% pivot_wider(names_from = "Condition", values_from = "Total_score")
EM_avg_wide <- EM_avg_wide %>% rename(Happy = `1`, Neutral = `2`)
Happy_mean <- mean(EM_avg_wide$Happy, na.rm = TRUE)
Happy_range <- range(EM_avg_wide$Happy, na.rm = TRUE)
Neutral_range <- range(EM_avg_wide$Neutral, na.rm = TRUE)
Neutral_mean <- mean(EM_avg_wide$Neutral, na.rm = TRUE)
happy_sd <- sd(EM_avg_wide$Happy, na.rm = TRUE)
neutral_sd <- sd(EM_avg_wide$Neutral, na.rm = TRUE)


#Performing t.test
results <- t.test(EM_avg_wide$Happy, EM_avg_wide$Neutral)
view(results)
print(results)
```

```
##
##  Welch Two Sample t-test
##
## data:  EM_avg_wide$Happy and EM_avg_wide$Neutral
## t = -0.089614, df = 166.46, p-value = 0.9287
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.833734  1.674497
## sample estimates:
## mean of x mean of y
##  21.41463  21.49425
```

```r
df <- results$parameter
print(df)
```

```
##       df
## 166.4626
```

```r
#Not significant
```

```r
#Converting GMAT test scores

newData$G.1 <-ifelse(grepl("Participation in art courses increases students creative thinking in their s

newData$G.2 <- ifelse(grepl("£117.00",newData$G.2), "2","0")

newData$G.3 <- ifelse(grepl("1-C, 2-A, 3-D, 4-B", newData$G.3), "2", "0")

newData$G.4 <- ifelse(grepl("S", newData$G.4), "2", "0")


correct_responses <- c("NBZSIEW", "IEWZNBS", "IEWSNBZ", "I E W Z S N B", "I, e, w, s, z, n, b", "IEWSZB
IWENBSZ", "IEWNBZS", "Z, B, N,W, I, E, S","N, B, S, Z, I, E, W","I E W N B Z S","I, E,W,Z,S,N,B","N Z B

newData$G.5 <- ifelse(grepl(paste(correct_responses, collapse = "|"), newData$G.5), 2, 0)
```

```r
newData$G.1 <- as.numeric(newData$G.1)
newData$G.2 <- as.numeric(newData$G.2)
newData$G.3 <- as.numeric(newData$G.3)
newData$G.4 <- as.numeric(newData$G.4)
newData$G.5 <- as.numeric(newData$G.5)

newData <- newData %>% rowwise() %>% mutate(Gmat_score = sum(c_across(21:25), na.rm = TRUE))
```

```r
#Newdata frames with Aggregate scores
final_df <- newData %>% select(prolificid, Condition, Current.mood, Age, Sex, Nationality, Ethnicity.si

final_df <- newData <- merge(EM_scores,final_df, by = c("prolificid", "Condition"), all = TRUE)
```

```r
#Checking MIP Efficacy
MIP <- final_df %>% select(Condition, Current.mood)

MIP <- MIP %>% mutate(CM_numeric = case_when(Current.mood == "Not at all happy" ~ 1, Current.mood == "N

MIP_H <- MIP$CM_numeric[MIP$Condition == "1"]
MIP_N <- MIP$CM_numeric[MIP$Condition == "2"]

t.test_MIP <- t.test(MIP_H, MIP_N)
print(t.test_MIP)
```

```
##
##  Welch Two Sample t-test
##
## data:  MIP_H and MIP_N
## t = 6.3003, df = 163.42, p-value = 2.647e-09
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.5264408 1.0070607
## sample estimates:
## mean of x mean of y
##  4.341463  3.574713
```

```r
view(t.test_MIP)
```

```r
#Calculating differences in GMAT scores

GMAT_scores <-  final_df %>% select(Condition, Gmat_score, prolificid)
GMAT_scores <- GMAT_scores %>% group_by(Condition)
GMAT_scores <- GMAT_scores %>% pivot_wider(names_from = "Condition", values_from = "Gmat_score")
GMAT_scores <- GMAT_scores %>% rename(Happy = `1`, Neutral = `2`)
results_Gmat <- t.test(GMAT_scores$Happy, GMAT_scores$Neutral)
view(results_Gmat)
GH_mean <- mean(GMAT_scores$Happy, na.rm = TRUE)
GN_mean <- mean(GMAT_scores$Neutral, na.rm = TRUE)
GH_sd <-  sd(GMAT_scores$Happy, na.rm = TRUE)
GN_sd <- sd(GMAT_scores$Neutral, na.rm = TRUE)

#not significant - no difference in scores of people in the happy vs. neutral condition
```

```r
#Happiness Scores
Happiness_score <- final_df %>% filter(Condition == "1")
```

```r
Happiness_counts <- table(Happiness_score$Current.mood)
print(Happiness_counts)
```

```
##
##           Extremely happy Neither happy nor unhappy            Not very happy
##                        38                         8                         1
##            Somewhat happy
##                        35
```

```r
Neutral_score <- final_df %>% filter(Condition == "2")
Neutral_counts <- table(Neutral_score$Current.mood)
print(Neutral_counts)
```

```
##
##           Extremely happy Neither happy nor unhappy            Not very happy
##                        16                        41                         6
##            Somewhat happy
##                        24
```

```r
#Compare EM scores of participants that reported feeling extremely happy in Condition 1, to Neither Hap
Current_Mood <- final_df %>% select(Condition, Current.mood, Total_score)
HP_hpymood <- Current_Mood %>% filter(Condition == 1, Current.mood == "Extremely happy") %>% select(Cur
NP_NtrlMood <- Current_Mood %>% filter(Condition == 2, Current.mood == "Neither happy nor unhappy") %>%
#Comparison via t-test
results_2 <- t.test(HP_hpymood$Total_score, NP_NtrlMood$Total_score)
mean_HP <- mean(HP_hpymood$Total_score, na.rm = TRUE)
mean_NP <- mean(NP_NtrlMood$Total_score, na.rm = TRUE)
sd_HP <- sd(HP_hpymood$Total_score, na.rm = TRUE)
sd_NP <- sd(NP_NtrlMood$Total_score, na.rm = TRUE)
print(results_2)
```

```
##
##  Welch Two Sample t-test
##
## data:  HP_hpymood$Total_score and NP_NtrlMood$Total_score
## t = 0.4296, df = 76.72, p-value = 0.6687
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.137394  3.313260
## sample estimates:
## mean of x mean of y
##  21.36842  20.78049
```

```r
view(results_2)
#No difference between people in Happiness condition that reported feeling happy later on and those in
```

```r
##Demographic Data Analysis
gender_counts <- table(final_df$Sex) #Gender
ethnicity_counts <- table(final_df$Ethnicity.simplified) #Ethnicity
Nationality_counts <- table(final_df$Nationality) #Nationality
Residency_counts <- table(final_df$Country.of.residence) #Residency
Student_counts <- table(final_df$Student.status) #Students

print(gender_counts)
```

```
##
```

```
## DATA_EXPIRED       Female         Male
##            1          118           50
```

```
print(ethnicity_counts)
```

```
##
##        Asian        Black DATA_EXPIRED        Mixed        Other        White
##            7           43            2            7            2          108
```

```
print(Nationality_counts)
```

```
##
##      Australia         Canada   DATA_EXPIRED          Ghana          India
##              2              4              1              1              1
##        Ireland    Netherlands    New Zealand        Nigeria       Pakistan
##              3              2              1              1              1
##   South Africa United Kingdom  United States       Zimbabwe
##             43            102              2              5
```

```
print(Residency_counts)
```

```
##
##      Australia         Canada        Germany        Ireland          Italy
##              3              4              1              3              1
##          Korea    New Zealand         Poland       Portugal   South Africa
##              1              3              1              1             48
## United Kingdom  United States
##            101              2
```

```
print(Student_counts)
```

```
##
## DATA_EXPIRED           No          Yes
##           25          106           38
```

```
#Analysing Gender Differences
Gender <- final_df %>% select(Sex, Condition, Total_score,prolificid)
Gender_H <- Gender %>% filter(Condition == "1")
Gender_H <- subset(Gender_H, select =  -2)
Gender_H <- Gender_H %>% pivot_wider(names_from = "Sex", values_from = "Total_score")
results_G <- t.test(Gender_H$Female, Gender_H$Male)

view(results_G)
print(results_G$parameter)
```

```
##       df
## 45.67646
```

```
Gender_N <- Gender %>% filter(Condition == "2")
Gender_N <- subset(Gender_N, select =  -2)
Gender_N <- Gender_N %>% pivot_wider(names_from = "Sex", values_from = "Total_score")
results_G_N <- t.test(Gender_N$Female, Gender_N$Male)
mean_gender <- t.test(Gender_H$Female,Gender_H$Male)
view(mean_gender)
view(results_G_N)
FH_mean <- mean(Gender_H$Female, na.rm = TRUE)
MH_mean <- mean(Gender_H$Male, na.rm = TRUE)
FH_sd <- sd(Gender_H$Female, na.rm = TRUE)
```

```r
MH_sd <- sd(Gender_H$Male, na.rm = TRUE)


#Analysing age differences
age_r <-  lm(Total_score ~ Age,final_df)
summary(age_r)
```

```
##
## Call:
## lm(formula = Total_score ~ Age, data = final_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.6667  -2.6667   0.5714   3.2000  11.6000
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   22.800      2.671   8.535 5.08e-14 ***
## Age21         -3.467      3.617  -0.958   0.3398
## Age22         -0.400      3.272  -0.122   0.9029
## Age23         -6.300      4.007  -1.572   0.1185
## Age24         -1.600      3.778  -0.424   0.6727
## Age25         -2.800      3.405  -0.822   0.4126
## Age26         -2.400      3.778  -0.635   0.5265
## Age27         -0.800      4.362  -0.183   0.8548
## Age28         -2.400      3.778  -0.635   0.5265
## Age29         -2.800      4.362  -0.642   0.5222
## Age30         -0.800      6.544  -0.122   0.9029
## Age31         -5.800      4.998  -1.161   0.2481
## Age32         -3.200      3.778  -0.847   0.3987
## Age33         -1.371      3.498  -0.392   0.6957
## Age34         -0.800      3.617  -0.221   0.8253
## Age35         -4.133      3.617  -1.143   0.2554
## Age36         -1.300      3.405  -0.382   0.7033
## Age37         -1.467      4.362  -0.336   0.7373
## Age38          1.200      3.778   0.318   0.7513
## Age39         -3.467      4.362  -0.795   0.4284
## Age40         -0.300      4.007  -0.075   0.9404
## Age41         -2.133      3.617  -0.590   0.5564
## Age42          1.200      4.007   0.299   0.7651
## Age43         -3.800      4.998  -0.760   0.4485
## Age44         -1.467      4.362  -0.336   0.7373
## Age45          6.200      4.007   1.547   0.1244
## Age46          3.200      6.544   0.489   0.6257
## Age48          2.200      4.998   0.440   0.6606
## Age49          0.200      4.007   0.050   0.9603
## Age50         -2.133      4.362  -0.489   0.6257
## Age51          0.200      4.007   0.050   0.9603
## Age52         -4.800      4.998  -0.960   0.3388
## Age53         -2.800      4.362  -0.642   0.5222
## Age54         -6.800      4.998  -1.361   0.1762
## Age55          4.200      4.007   1.048   0.2967
## Age56          4.200      4.998   0.840   0.4024
## Age57         -1.800      4.998  -0.360   0.7194
```

```
## Age58              4.200      4.998   0.840   0.4024
## Age60              5.200      6.544   0.795   0.4284
## Age61             -1.800      4.998  -0.360   0.7194
## Age62             -1.800      4.998  -0.360   0.7194
## Age63             -1.800      4.998  -0.360   0.7194
## Age65             -2.800      6.544  -0.428   0.6695
## Age66             -9.800      4.998  -1.961   0.0522 .
## Age67              5.200      6.544   0.795   0.4284
## Age68             -6.800      6.544  -1.039   0.3008
## Age69              1.200      6.544   0.183   0.8548
## Age74             -2.800      6.544  -0.428   0.6695
## AgeDATA_EXPIRED    1.200      6.544   0.183   0.8548
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.974 on 120 degrees of freedom
## Multiple R-squared:  0.2307, Adjusted R-squared:  -0.07705
## F-statistic: 0.7496 on 48 and 120 DF,  p-value: 0.8706
```

```
#Difference in scores in women in experimental group vs. control group
fvf <- t.test(Gender_H$Female,Gender_N$Female)
happy_f_mean <- mean(Gender_H$Female, na.rm = TRUE)
neutral_f_mean <- mean(Gender_N$Female, na.rm = TRUE)
mvm <- t.test(Gender_H$Male,Gender_N$Male)
view(mvm)
view(fvf)
```