Seeing is Believing

Jason Geller

Last Knitted: 2023-09-18

### Vignette Setup:

knitr::opts\_chunk$set(echo = TRUE)  
  
# Libraries necessary for this vignette  
library(rio)  
library(flextable)  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(tidyr)  
library(semanticprimeR)  
set.seed(92747)  
  
# Function for simulation  
item\_power <- function(data, # name of data frame  
 dv\_col, # name of DV column as a character  
 item\_col, # number of items column as a character  
 sample\_start = 20,   
 sample\_stop = 200,   
 sample\_increase = 5,  
 decile = .5){  
   
 DF <- cbind.data.frame(  
 dv = data[ , dv\_col],  
 items = data[ , item\_col]  
 )  
   
 # just in case  
 colnames(DF) <- c("dv", "items")  
   
 # figure out the "sufficiently narrow" ci value  
 SE <- tapply(DF$dv, DF$items, function (x) { sd(x)/sqrt(length(x)) })  
 cutoff <- quantile(SE, probs = decile)  
   
 # sequence of sample sizes to try  
 samplesize\_values <- seq(sample\_start, sample\_stop, sample\_increase)  
  
 # create a blank table for us to save the values in   
 sim\_table <- matrix(NA,   
 nrow = length(samplesize\_values),   
 ncol = length(unique(DF$items)))  
  
 # make it a data frame  
 sim\_table <- as.data.frame(sim\_table)  
  
 # add a place for sample size values   
 sim\_table$sample\_size <- NA  
  
 # loop over sample sizes  
 for (i in 1:length(samplesize\_values)){  
   
 # temp that samples and summarizes  
 temp <- DF %>%   
 group\_by(items) %>%   
 sample\_n(samplesize\_values[i], replace = T) %>%   
 summarize(se = sd(dv)/sqrt(length(dv)))  
   
 # dv on items  
 colnames(sim\_table)[1:length(unique(DF$items))] <- temp$items  
 sim\_table[i, 1:length(unique(DF$items))] <- temp$se  
 sim\_table[i, "sample\_size"] <- samplesize\_values[i]  
   
 }  
  
 # figure out cut off  
 final\_sample <- sim\_table %>%   
 pivot\_longer(cols = -c(sample\_size)) %>%   
 dplyr::rename(item = name, se = value) %>%   
 group\_by(sample\_size) %>%   
 summarize(percent\_below = sum(se <= cutoff)/length(unique(DF$items)))   
   
 # multiply by correction   
 final\_sample$new\_sample <- round(39.369 + 0.700\*final\_sample$sample\_size + 0.003\*cutoff - 0.694\*length(unique(DF$items)))  
   
 return(list(  
 SE = SE,   
 cutoff = cutoff,   
 DF = DF,   
 sim\_table = sim\_table,   
 final\_sample = final\_sample  
 ))  
  
}

### Project/Data Title:

Seeing Is Believing: How Media Type Effects Truth Judgements

Data provided by: Gianni Ribeiro

### Project/Data Description:

People have been duly concerned about how fake news influences the minds of the populous since the rise of propaganda in World War One (Lasswell, 1927). Experts are increasingly worried about the effects of false information spreading over the medium of video. Members of the deep trust alliance, a global network of scholars researching deepfakes and doctored videos, state that ‘a fundamental erosion of trust is already underway’ (Harrison, 2020). Newman et al. (2015) discovered that the media type through which information is presented does indeed affect how true the information feels. Newman speculated that this truthiness effect could be because images provide participants with more information than text alone, thus making the source feel more informationally rich.

In this experiment, our aim is to test the generalizability of Newman’s truthiness effect in two ways: first, to see if it extends to other media types in addition to images, and second, to test if it applies to other domains. In this study, we will present individuals with true and false claims presented through three different media types: (1) text, (2) text alongside a photo, and (3) text alongside a video. This is a direct replication of Newman’s experiment, just with the addition of the video condition. Similarly, participants will also be asked to make truth judgements about trivial claims and claims about COVID-19, to see if the truthiness effect extends to other domains besides trivia.

In this within-subjects design, participants will be presented with true and false claims about trivia and COVID-19 in counterbalanced order. These claims will be randomly assigned to appear either as text alone, text alongside an image, or text alongside a video. Participants will be asked to rate how true they believe each claim is.

### Methods Description:

Participants were largely sourced from the first-year participant pool at The University of Queensland. Participation was completely voluntary, and participants can choose to withdraw at any time.

Thirty matched trivia claims were generated directly from Newman’s materials. These claims were selected and a true and false version of each claim was created. Newman’s original claims are available at the following link: <https://data.mendeley.com/datasets/r68dcdjrpc/1>

The second set of materials comprising of matched true and false claims was generated using information resources from the World Health Organisation, and various conspiracy websites. These claims were then fact-checked by Kirsty Short, an epidemiologist and senior lecturer in the School of Chemistry and Molecular Sciences at The University of Queensland.

The claims were also pilot tested to ensure that none of them performed at floor or ceiling. This pilot test consisted of 56 participants and subsequently four claims were dropped. The data from this pilot test was also used to accurately perform a power analysis. After generating the means of the pilot test, we found that to acquire a power of 0.8 or greater, there must be a mean difference of 0.4 between each media type. This mean difference is quite conservative, since we plan to measure truth ratings on a six-point scale and is easily achievable with 100 participants.

The videos were largely sourced from the stock image website Envato Elements and Screenflow’s Royalty Free Stock Media Library.

### Data Location:

Data can be found here: <https://osf.io/zu9pg/>

DF <- import("moat\_data.csv")   
   
str(DF)

## 'data.frame': 5040 obs. of 11 variables:  
## $ id : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ domain : chr "covid" "covid" "covid" "covid" ...  
## $ gender : chr "Female" "Female" "Female" "Female" ...  
## $ age : int 22 22 22 22 22 22 22 22 22 22 ...  
## $ filename : chr "1\_kaitlin\_exp\_v2b\_97832952686.txt" "1\_kaitlin\_exp\_v2b\_97832952686.txt" "1\_kaitlin\_exp\_v2b\_97832952686.txt" "1\_kaitlin\_exp\_v2b\_97832952686.txt" ...  
## $ medium : chr "claim" "claim" "claim" "claim" ...  
## $ question\_type : chr "drinking" "herd" "hydroxychloroquine" "steam" ...  
## $ truth : logi TRUE TRUE TRUE TRUE TRUE FALSE ...  
## $ claim : chr "COVID-19 cannot be contracted through drinking water." "Herd immunity against COVID-19 cannot be achieved by letting the virus spread through the population." "Studies show hydroxychloroquine does not have clinical benefits in treating COVID-19." "Steam inhalation cannot help cure COVID-19." ...  
## $ filename\_other: chr NA NA NA NA ...  
## $ rating : int 1 5 6 6 6 1 1 1 5 1 ...

### Date Published:

No official publication, see citation below.

### Dataset Citation:

Moat, K., Tangen, J., & Newman, E. (2021). Seeing Is Believing: How Media Type Effects Truth Judgements.

### Keywords:

covid, trivia games, claims, interpretation

### Use License:

Open access with reference to original paper (Attribution-NonCommercial-ShareAlike CC BY-NC-SA)

### Geographic Description - City/State/Country of Participants:

Brisbane, Queensland, Australia

### Column Metadata:

metadata <- tibble::tribble(  
 ~Variable.Name, ~Variable.Description, ~`Type.(numeric,.character,.logical,.etc.)`,  
 "Id", "Participant ID", "numeric",  
 "Domain", "Whether the trial is a claim about COVID ('covid') or TRIVIA ('trivia)", "character",  
 "Medium", "Whether the trial appears as text alone ('claim'), text alongside an image ('photo'), or text alongside a video ('video')", "character",  
 "Trial\_type", "Whether the trial presents a claim that is TRUE ('target') or FALSE ('distractor')", "character",  
 "Rating", "Paritcipant’s truth rating of the claim ranging from 1 (definitely false) to 6 (definitely tue)", "numeric"  
 )  
  
flextable(metadata) %>% autofit()

| Variable.Name | Variable.Description | Type.(numeric,.character,.logical,.etc.) |
| --- | --- | --- |
| Id | Participant ID | numeric |
| Domain | Whether the trial is a claim about COVID ('covid') or TRIVIA ('trivia) | character |
| Medium | Whether the trial appears as text alone ('claim'), text alongside an image ('photo'), or text alongside a video ('video') | character |
| Trial\_type | Whether the trial presents a claim that is TRUE ('target') or FALSE ('distractor') | character |
| Rating | Paritcipant’s truth rating of the claim ranging from 1 (definitely false) to 6 (definitely tue) | numeric |

### AIPE Analysis:

# Function for simulation  
var1 <- item\_power(data = DF, # name of data frame  
 dv\_col = "rating", # name of DV column as a character  
 item\_col = "question\_type", # number of items column as a character  
 sample\_start = 20,   
 sample\_stop = 300,   
 sample\_increase = 5,  
 decile = .4)

#### Stopping Rule

What is the usual standard error for the data that could be considered for our stopping rule?

var1$SE

## afghan antibiotics bacteria blindness   
## 0.1733704 0.2192951 0.2550577 0.1787244   
## breastfeeding chemical children cocaine   
## 0.1726203 0.2000046 0.2224147 0.2427085   
## colorvision corn couscous curling   
## 0.1284245 0.1064393 0.1337200 0.1500109   
## dartboards denmark dna drinking   
## 0.1322563 0.1802086 0.2273651 0.1732374   
## elderly fishing forest foxhunting   
## 0.2576951 0.1464690 0.1349403 0.1446861   
## grapes herd hiv houseflies   
## 0.1573876 0.1877676 0.2135034 0.1907986   
## hydroxychloroquine infertility lawnbowls lime   
## 0.1495606 0.2049964 0.1147161 0.1465564   
## longbow marathon microwave mintonette   
## 0.1667281 0.1392743 0.1959902 0.1066397   
## mosquitoes mountains mouthwash nile   
## 0.1894154 0.0912085 0.2245578 0.1532541   
## otter oxygen oysters penicillin   
## 0.1280050 0.2093482 0.1173648 0.1584095   
## poland prisoners rate rigor   
## 0.1326624 0.1970849 0.2147593 0.2238266   
## saline smell snake snowboarding   
## 0.2132273 0.2432145 0.1899915 0.1359677   
## steam temperature triathalon twice   
## 0.2243410 0.2215570 0.1851070 0.2292498   
## urchin uv vesuvius vitamin-c   
## 0.1399591 0.2161739 0.1433344 0.2127463   
## vitamin-d water waterfall zulu   
## 0.2251918 0.1817230 0.1411920 0.1264954

var1$cutoff

## 40%   
## 0.1580007

cutoff <- var1$cutoff  
  
# we can also use semanticprimer's function  
cutoff\_score <- calculate\_cutoff(population = DF,  
 grouping\_items = "question\_type",  
 score = "rating",  
 minimum = min(DF$rating),  
 maximum = max(DF$rating))  
cutoff\_score$cutoff

## 40%   
## 0.1580007

Using our 40% decile as a guide, we find that 0.158 is our target standard error for an accurately measured item.

#### Minimum Sample Size

To estimate minimum sample size, we should figure out what number of participants it would take to achieve 80%, 85%, 90%, and 95% of the SEs for items below our critical score of 0.158?

How large does the sample have to be for 80% of the items to be below our stopping SE rule?

flextable(var1$final\_sample %>% head()) %>%   
 autofit()

| sample\_size | percent\_below | new\_sample |
| --- | --- | --- |
| 20 | 0.00000000 | 12 |
| 25 | 0.01666667 | 15 |
| 30 | 0.05000000 | 19 |
| 35 | 0.05000000 | 22 |
| 40 | 0.06666667 | 26 |
| 45 | 0.03333333 | 29 |

final\_table <- calculate\_correction(  
 proportion\_summary = var1$final\_sample,  
 pilot\_sample\_size = DF %>% group\_by(question\_type) %>%   
 summarize(sample\_size = n()) %>% ungroup() %>%   
 summarize(avg\_sample = mean(sample\_size)) %>% pull(avg\_sample),  
 proportion\_variability = cutoff\_score$prop\_var  
 )  
  
flextable(final\_table) %>%   
 autofit()

| percent\_below | sample\_size | corrected\_sample\_size |
| --- | --- | --- |
| 80.00000 | 155 | 93.0211 |
| 86.66667 | 175 | 104.8176 |
| 93.33333 | 180 | 107.9577 |
| 95.00000 | 195 | 116.7822 |

Based on these simulations, we can decide our minimum sample size is likely close to 93.

#### Maximum Sample Size

In this example, we could set our maximum sample size for 90% of items, which would equate to 108 participants.

#### Final Sample Size

You should also consider any potential for missing data and/or unusable data given the requirements for your study. Given that participants are likely to see all items in this study, we could use the minimum, stopping rule, and maximum defined above.