959_replication

Dasha Castillo

2023-05-11

9.59 Replication Project

Below, we'll be loading the data of the replication of the Wu & Gibson (2021) experiment.

200 participants were able to take the survey for the experiment.

```
# loading data for "english replication" aka eng_rep
en_rep <- read_csv("https://raw.githubusercontent.com/dashacastillo/959_wu_gibson_replication/main/9.59
```

```
## Rows: 830 Columns: 10
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (8): participant_id, experiment, cond, literal_response, response, times...
## dbl (2): item, presentation_order
##
## 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.
```

```
en_rep$response = tolower(en_rep$response)
en_rep
```

```
## # A tibble: 830 x 10
##    partici~1 exper~2  item cond  prese~3 liter~4 respo~5 times~6 engli~7 count~8
##    <chr>     <chr>   <dbl> <chr>   <dbl> <chr>   <chr>   <chr>   <chr>   <chr>
##  1 60fedd38~ color       1 bott~       0 Yes     purple~ 5/10/2~ Yes     United~
##  2 60fedd38~ color       2 top_~       1 No      yellow~ 5/10/2~ Yes     United~
##  3 60fedd38~ color       4 bott~       2 No      purple~ 5/10/2~ Yes     United~
##  4 60fedd38~ color       3 top_~       3 Yes     pink s~ 5/10/2~ Yes     United~
##  5 5dc985b8~ color       1 top_~       0 Yes     square  5/10/2~ Yes     United~
##  6 5dc985b8~ color       2 bott~       1 Yes     square  5/10/2~ Yes     United~
##  7 5dc985b8~ color       3 bott~       2 No      triang~ 5/10/2~ Yes     United~
##  8 5dc985b8~ color       4 top_~       3 No      rectan~ 5/10/2~ Yes     United~
##  9 5c770b45~ color       3 bott~       0 No      yellow~ 5/10/2~ Yes     United~
## 10 5c770b45~ color       1 bott~       1 Yes     purple~ 5/10/2~ Yes     United~
## # ... with 820 more rows, and abbreviated variable names 1: participant_id,
## #   2: experiment, 3: presentation_order, 4: literal_response, 5: response,
## #   6: timestamp, 7: english_first_language, 8: country_of_birth
```

```
# loading data for "spanish replication" aka sp_rep
sp_rep <- read_csv("https://raw.githubusercontent.com/dashacastillo/959_wu_gibson_replication/main/9.59
```

```
## Rows: 832 Columns: 10
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (7): participant_id, experiment, cond, literal_response, response, engl...
## dbl  (2): item, presentation_order
## dttm (1): timestamp
##
## 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.
```

```
sp_rep$response = tolower(sp_rep$response)
sp_rep
```

```
## # A tibble: 832 x 10
##    participant~1 exper~2  item cond  prese~3 liter~4 respo~5 timestamp
##    <chr>         <chr>   <dbl> <chr>   <dbl> <chr>   <chr>   <dttm>
##  1 5f3f1ae9db53~ color       2 top_~       0 No      rectan~ 2023-05-10 19:45:48
##  2 5f3f1ae9db53~ color       4 bott~       1 Yes     corazo~ 2023-05-10 19:45:48
##  3 5f3f1ae9db53~ color       3 bott~       2 No      triang~ 2023-05-10 19:45:48
##  4 5f3f1ae9db53~ color       1 top_~       3 Yes     cuadra~ 2023-05-10 19:45:48
##  5 545d37b5fdf9~ color       2 bott~       0 No      círcul~ 2023-05-10 19:46:00
##  6 545d37b5fdf9~ color       3 top_~       1 No      rectán~ 2023-05-10 19:46:00
##  7 545d37b5fdf9~ color       1 top_~       2 Yes     cuadro~ 2023-05-10 19:46:00
##  8 545d37b5fdf9~ color       4 bott~       3 Yes     corazó~ 2023-05-10 19:46:00
##  9 5f9baf2a2613~ color       1 bott~       0 No      triáng~ 2023-05-10 19:46:03
## 10 5f9baf2a2613~ color       4 top_~       1 Yes     rombo   2023-05-10 19:46:03
## # ... with 822 more rows, 2 more variables: english_first_language <chr>,
## #   country_of_birth <chr>, and abbreviated variable names 1: participant_id,
## #   2: experiment, 3: presentation_order, 4: literal_response, 5: response
```

Data Organization

The main goal of the experiment is to see the redundancy in describing the shapes. One way to see whether or not a redundant describer was given is to find all of the unique values of the responses and then making a new colum, Redundant, in which a value of 1 refers to whether a redundant modification was used, and a value of 0 refers to a non-redundant description.

```
# all unique english descriptions from the responses
en_descs <- unique(en_rep$response)

en_red <- en_descs[c(1:4, 8:22, 29, 30, 36, 40:46, 48, 49, 54, 55, 58, 61:65, 67:70,
                     72, 73, 76:79, 83, 84, 91:94, 96, 99:115, 118:126, 128)]

en_nonred <- en_descs[c(5:7, 24:28, 32:35, 38, 39, 47, 50:53, 56, 57, 59, 60, 66, 71,
                        74, 75, 80:82, 85:90, 95, 97, 98, 117, 127, 133:138)]

en_throwaway <- en_descs[c(23, 31, 129:132)]

en_unsure <- en_descs[c(37, 116)]
```

```
# all unique spanish descriptions from the responses
sp_descs <- unique(sp_rep$response)
```

```
sp_red <- sp_descs[c(1:8, 15, 16, 21, 24:28, 31:35, 37:53, 58:71, 74, 76:81, 83, 86:95,
                     100, 103:110, 113:119, 121:125, 127:131, 134:139)]

sp_nonred <- sp_descs[c(9:14, 17:20, 22, 23, 29, 30, 36, 54:57, 72, 73, 75, 82, 84, 85,
                        96:99, 101, 102, 120, 126, 132, 133)]

sp_throwaway <- sp_descs[c()]

sp_unsure <- sp_descs[c(111, 112)]
```

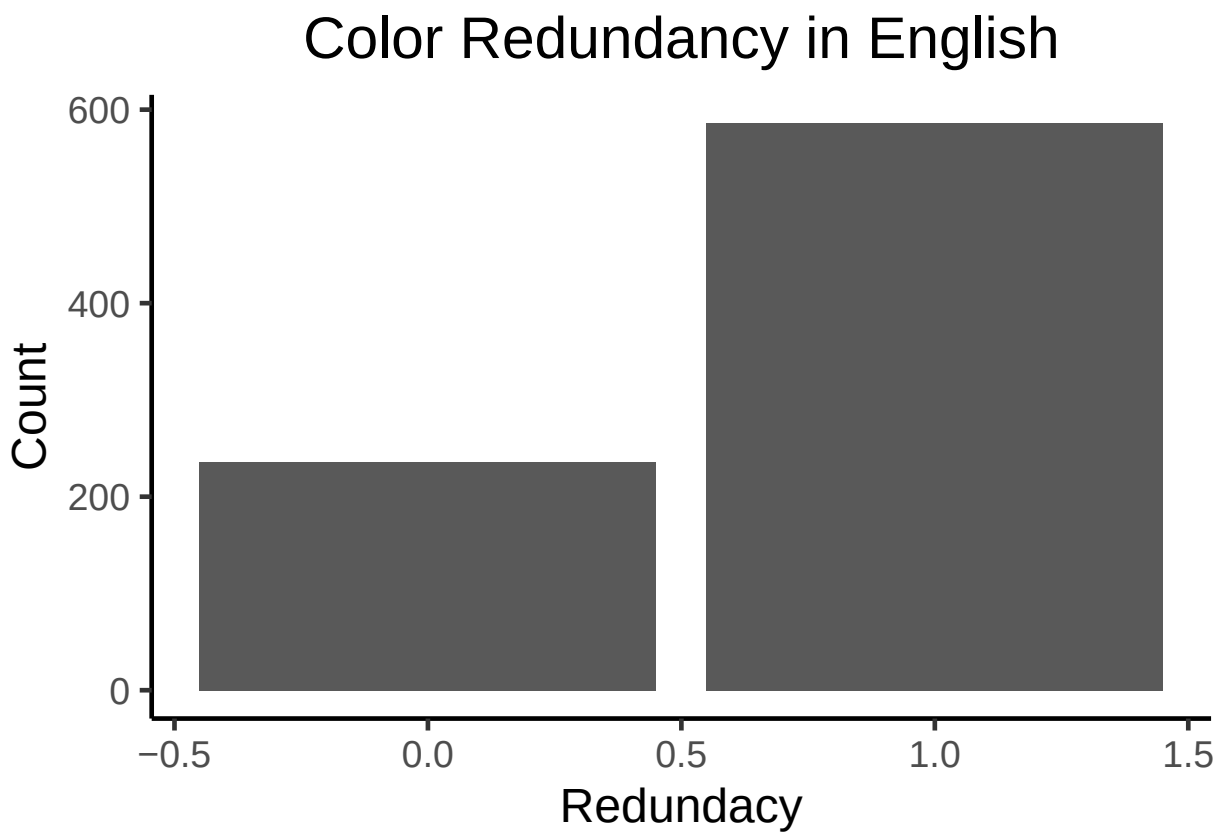
Adding a New Column to Show if Redundancy Exists in Response

```
# first delete the unsure and throwaway items!
# we can't analyze what we can't put in a binary

en_replication_data = subset(en_rep,
                             response!="sub" &
                               response!="marks the spot" &
                               response!="1" & response!="2" &
                               response!="4" & response!="3" &
                               response!="plus, cross" &
                               response!="normal cirlcle")
# summary(en_replication_data)

en_rep_redundancy = mutate(en_replication_data, redundant = if_else(en_replication_data$response %in% e
# test to see
# en_rep_redundancy[c("response", "redundant")]
```

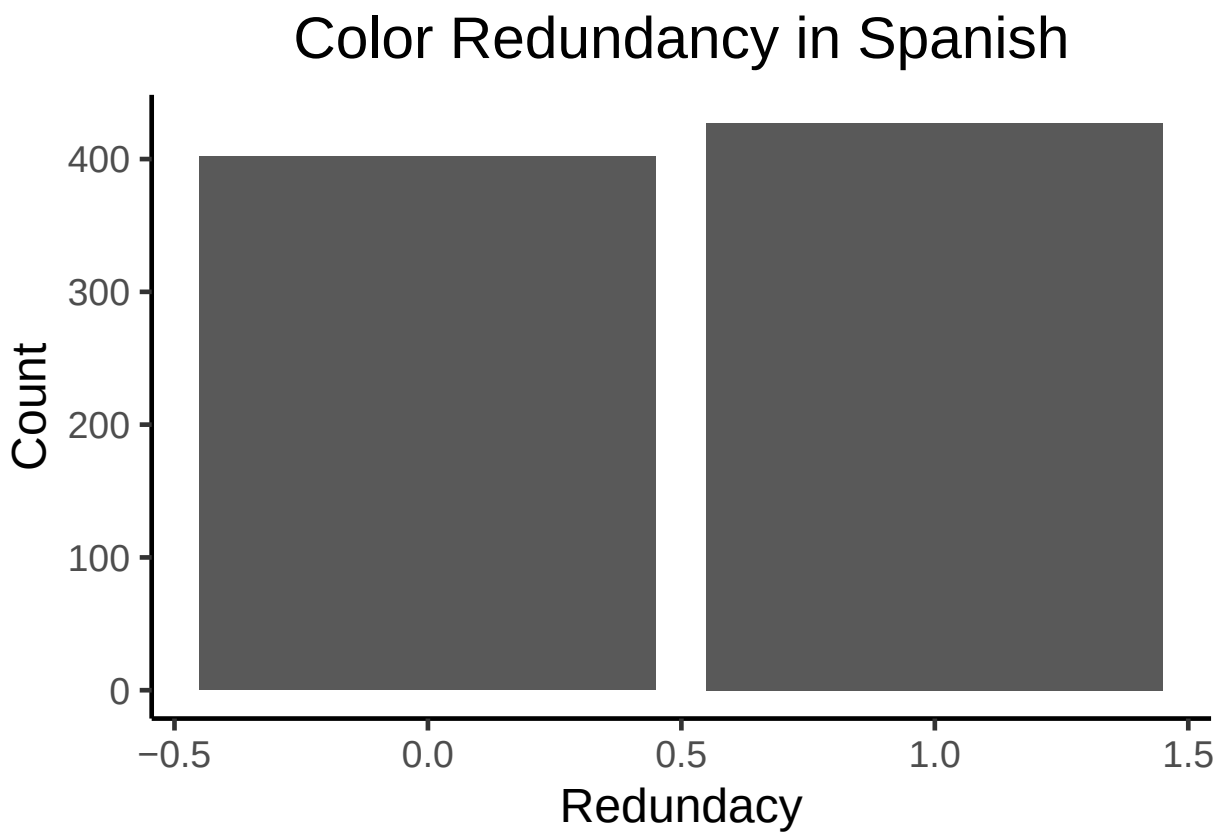
```
# first delete the unsure and throwaway items!
# we can't analyze what we can't put in a binary

sp_replication_data = subset(sp_rep,
                             response!="más morado" &
                               response!="más")
# summary(sp_replication_data)

sp_rep_redundancy = mutate(sp_replication_data, redundant = if_else(sp_replication_data$response %in% s
# test to see
# sp_rep_redundancy[c("response", "redundant")]
```

Making Figures

```
ggplot(en_rep_redundancy, aes(x = redundant)) +
  geom_bar() +
  ggtitle("Color Redundancy in English") +
  xlab("Redundacy") + ylab("Count")
```




```
ggplot(sp_rep_redundancy, aes(x = redundant)) +
  geom_bar() +
  ggtitle("Color Redundancy in Spanish") +
  xlab("Redundacy") + ylab("Count")
```


Color Redundancy in Spanish
Count
400
300
200
100
0
−0.5
0.0
0.5
1.0
1.5
Redundacy