Short Report 2

Parker Johnson and Teagan Johnson

2/15/2022

Introduction

One approach to measure somebody's understanding of the English language is to determine how many English words they know. This, however, seems very difficult to determine without any analysis. In this study, we estimate the number of words an individual knows in Webster's dictionary with a one-stage cluster design in an effort to figure out the breadth of their English vocabulary.

Methodology

For this design, we applied a one-stage cluster sample sampling design. This design seemed appropriate for our study, because cluster sampling by pages is convenient and time-effective as opposed to finding random words across the dictionary. Additionally, there are a large number of pages and words per page in this dictionary, so normality assumptions were met, and each page consists of different words, so the samples are non-overlapping.

The population of observation units was the total number of words in the dictionary which is unknown, and the sampling units (clusters) in our one-stage cluster design were the pages. The total number of clusters in our population was 1527, and our sampling frame was all 1527 pages.

Our sampling design consisted of us taking a random sample of pages (clusters). First, we needed to choose an appropriate sample size for our design. We decided to aim for a margin of error of 5% for our estimates of the total words Parker knows and the proportion of words Parker knows in the dictionary. We assume, conservatively, that the proportion of words Parker knows per page is 0.5, which yields a standard deviation of words Parker knows per page of 0.5. Using the simple random sample size calculation and applying the finite population correction, we found that we should sample 307 words to achieve a margin of error of 5%.

However, we still needed to find the number of pages we should sample since the current sample size is in terms of words. To do this, we took a pilot sample of 5 pages from the dictionary to find a rough estimate of the average number of words per page. This was found to be roughly 50 words per page. Therefore, the number of pages we needed to sample was around $307 / 50 \approx 6.07$ pages, which we rounded up to 7 pages to still capture our goal margin of error of 5%.

From this, we were able to conduct our survey by randomly selecting 7 pages from the dictionary, and counting the total number of words on the page and all of the words on the page that Parker knew. To be consistent with our measurements, we needed to precisely define what it means to "know" a word. A word was determined to be known if Parker had seen the word before, and had a rough understanding of how to use the word in a sentence. After recording our data in a dataframe, we used the survey package in R to calculate the estimated total number of words that Parker knows in the dictionary. Then we calculated the estimated proportion of the words that Parker knows in the dictionary and their corresponding confidence intervals. The estimated proportion of the words that Parker knows was obtained with the biased ratio estimator because the total number of words in the dictionary is not known.

Results

The results from each of the 7 cluster samples can be seen in Figure 1. The largest number of words on a page was 53 words, which also corresponded to the largest number of words Parker knew on a page, which was 41 words. The smallest number of words on a page was 47 words, and the smallest number of words Parker knew on a page was 35 words. A table of the results can be found at the end of the appendix.

We found that the total number of words Parker Johnson knew was estimated to be 58244 with a standard error of 1502.6. We are 95% confident that the true number of words Parker knew in the dictionary is between 54567.35 and 61920.93. The proportion of words that Parker knew was estimated to be 0.76 with a standard error of 0.02. We are 95% confident that the true proportion of words in the dictionary that Parker knew is between 0.71 and 0.82.

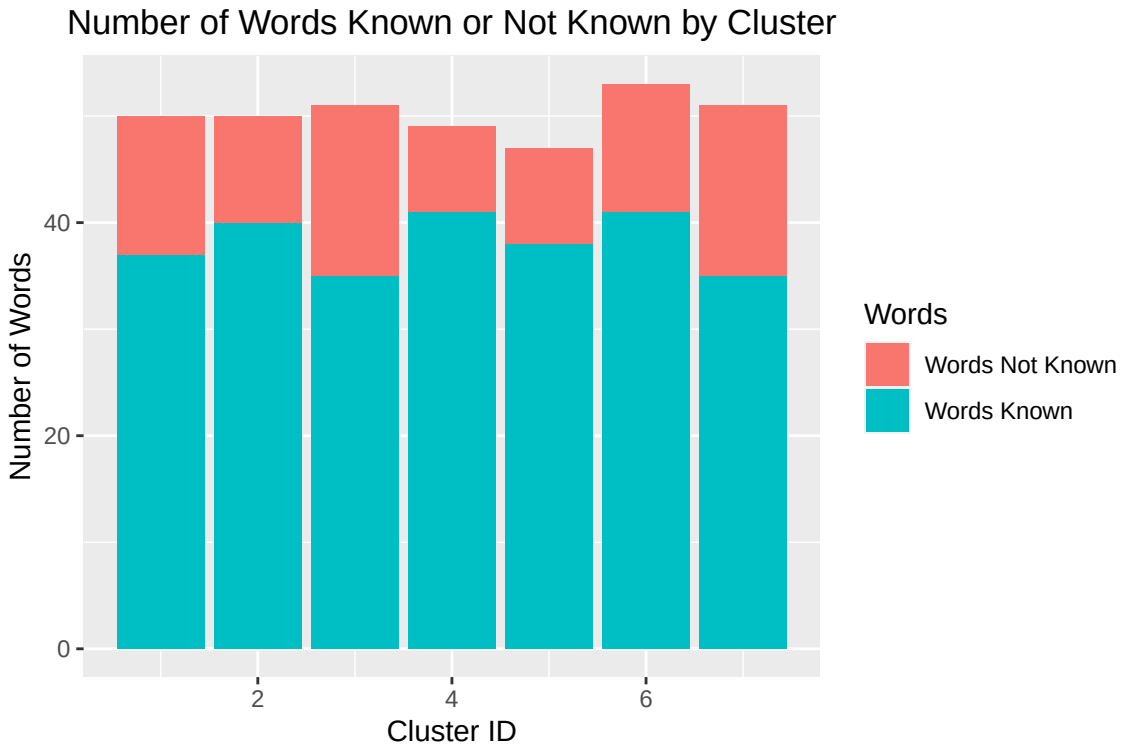


Figure 1: Stacked bar graph showing proportions of words known in each of our samples.

Discussion

We estimate that Parker knows roughly 58244 words from the dictionary, or 76% of all the words from the dictionary using a one-stage cluster design. However, one-stage cluster designs come with drawbacks.

The main drawback of the one-stage cluster design is that it has a lower precision. This is due to the fact that clusters often have heterogeneous responses between clusters and their responses are homogeneous within clusters. In the case of our dictionary design, there is potential for both. Pages may have different average responses based on the beginning of words (i.e words that start with "ba-" might be more widely-understood than words that start with "xy-"). Because of this loss of precision in conducting a cluster sample compared to a simple random sample, our margin of error of 5% was not achieved, which can be seen in our confidence interval for the proportion of words Parker knows spanning more than 10%.

One source of bias is that Parker could have purposefully said he knew more words on a page than he actually did to show that he knew more words. This would skew all of the responses to be larger than they actually are, resulting in bias.

It's possible that there were some measurement errors in our study. One such example is determining whether or not Parker knew a word. Although we attempted to provide a strict definition of whether or not Parker knows a word, there is still lots of gray area between a full understanding and no understanding of a word. Another potential source of measurement error is our estimation of the number of words per page. In order to calculate the cluster sample size, we estimated the number of words per page with a pilot sample of 5 pages. It's possible we didn't sample enough pages to produce a reliable estimation of words per page, which would impact our data.

Although our sample size of clusters is relatively small, our population responses are normally distributed. Therefore, we assumed that the assumptions (clusters sampled randomly and independently, and sample size is less than 10% of population) for confidence intervals were met.

Appendix The first step in our sampling design is to find the cluster sample size n. We did this using the SRS sample size formula $n = \frac{n_0}{1+\frac{n_0}{N}}$ where $n_0 = (\frac{sz}{e})^2$ to find the number of words we needed to sample. Then we divided this by an estimate of the number of words per page (determined with a pilot sample of 5 pages).

```
# First we find the sample size of words
s <- .5
e <- .05
z <- 1.96
N <- 1527
n_0 <- (s*z/e)^2
n <- n_0/(1+(n_0/N))
n
```

```
## [1] 306.9405
```

```
# Pilot sample to determine a rough estimate of the number of words per page
set.seed(754332)
sample(1:N, 7)
```

```
## [1] 828 778 917 177 907 338 796
```

```
words_per_page <- round(mean(c(51, 49, 53, 51, 45))) # We found the mean number
                      #of words per page to be 49.8. We round this up to be 50

n <- n/words_per_page # Round up to 7, this is the sample size
n <- 7
```

Once we found the cluster sample size, we took a simple random sample of 7 pages in the dictionary. Parker responded with how many words he knew on each page and how many words were on each page.

```
# Random sample of 7 pages
set.seed(47352948)
sample(1:N, 7)
```

```
## [1]  293 1254  430  169  583 1184  884
```

```
#Page 293: out of 50 words, Parker knows 37
#Page 1254: out of 50 words, Parker knows 40
#Page 430: out of 51 words, Parker knows 35
#Page 169: out of 49 words, Parker knows 41
#Page 583: out of 47 words, Parker knows 38
#Page 1184: out of 53 words, Parker knows 41
#Page 884: out of 51 words, Parker knows 35
```

Using the known vs. total words per page, we created a cluster-level data frame (each row represents a cluster) with 3 columns: cluster_id, words_known, total_words.

```r
# Now we'll create a data frame
cluster_id <- c(1, 2, 3, 4, 5, 6, 7)
words_known <- c(37, 40, 35, 41, 38, 41, 35)
total_words <- c(50, 50, 51, 49, 47, 53, 51)
dict_f <- data.frame(cluster_id, words_known, total_words)
dict_f
```

```
##   cluster_id words_known total_words
## 1          1          37          50
## 2          2          40          50
## 3          3          35          51
## 4          4          41          49
## 5          5          38          47
## 6          6          41          53
## 7          7          35          51
```

Now that our data frame is constructed, we're ready to calculate our estimates. First, we set up a survey design object with our data frame. Then, we use svytotal from the survey package to estimate the total number of words in the dictionary Parker knows. Next, we use svyratio to estimate the proportion of words in the dictionary Parker knows.

```r
# Estimation
library(survey)
```

```
## Loading required package: grid
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
## Loading required package: survival
##
## Attaching package: 'survey'
## The following object is masked from 'package:graphics':
##
##     dotchart
```

```r
# Add population, sample sizes to each row to construct survey design object.
dict_f$N <- N
dict_f$n <- n
dict_f$wts <- dict_f$N/dict_f$n

dict_design <- svydesign(id=~1, fpc=~N, weights=~wts, data=dict_f)

# Total
svytotal(~words_known, dict_design)
```

```
##             total     SE
## words_known 58244 1502.6
```

```r
confint(svytotal(~words_known, dict_design), df=degf(dict_design))
```

```
##                2.5 %   97.5 %
```

```
## words_known 54567.35 61920.93
```

```
# Proportion
svyratio(~words_known, ~total_words, dict_design)
```

```
## Ratio estimator: svyratio.survey.design2(~words_known, ~total_words, dict_design)
## Ratios=
##             total_words
## words_known   0.7606838
## SEs=
##             total_words
## words_known  0.02238168
```

```
confint(svyratio(~words_known, ~total_words, dict_design), df=degf(dict_design))
```

```
##                               2.5 %    97.5 %
## words_known/total_words 0.7059178 0.8154498
```

Below we calculate the design effect comparing the cluster variance to a baseline SRS variance.

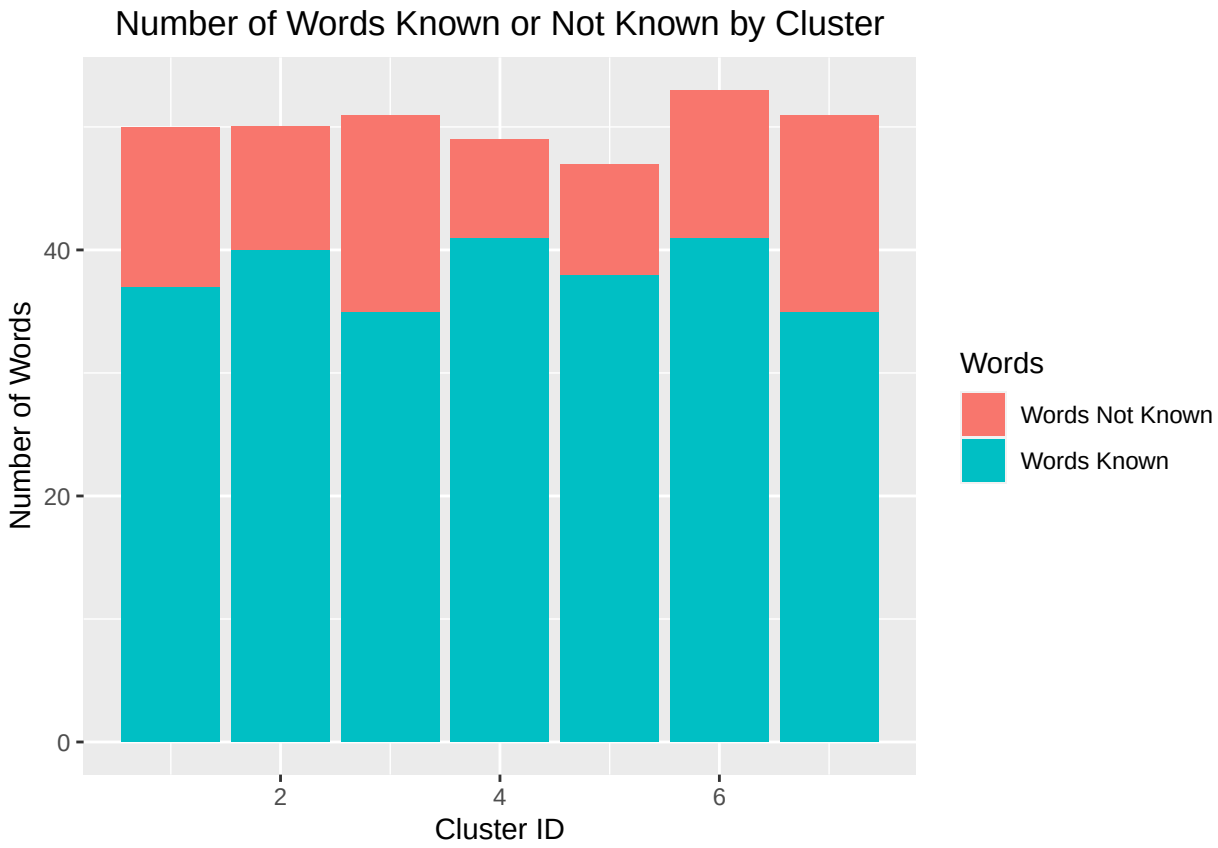
```
# DEff
cluster_var <- 0.02238168^2
SRS_var <- ((1 - (307/(50*1527))) * (0.7606838*(1-0.7606838)) / (307-1))
cluster_var/SRS_var
```

```
## [1] 0.8454352
```

Now we create the stacked bar graph of our sampled data.

```
# EDA
library(ggplot2)
library(tidyverse)
dict_f$words_not_known <- (dict_f$total_words - dict_f$words_known)
dict_f_test <- dict_f %>%
  select("cluster_id", "words_known", "words_not_known") %>%
  mutate("Words Known" = words_known, "Words Not Known" = words_not_known) %>%
  select("cluster_id", "Words Known", "Words Not Known")

graph <- dict_f_test %>%
    pivot_longer(-cluster_id, names_to = "words",
                 values_to = "Number of Words") %>%
    ggplot(aes(cluster_id, `Number of Words`, fill= forcats::fct_rev(words))) +
    geom_col() +
    labs(x = "Cluster ID", y = "Number of Words", fill = "Words", title =
           "Number of Words Known or Not Known by Cluster") +
    theme(plot.title = element_text(hjust = 0.5))
graph
```



Below is our data frame. The "words_not_known" column was used to create our stacked bar plot.

```
knitr::kable(dict_f)
```

cluster_id	words_known	total_words	N	n	wts	words_not_known
1	37	50	1527	7	218.1429	13
2	40	50	1527	7	218.1429	10
3	35	51	1527	7	218.1429	16
4	41	49	1527	7	218.1429	8
5	38	47	1527	7	218.1429	9
6	41	53	1527	7	218.1429	12
7	35	51	1527	7	218.1429	16