POL290G Module 1 HW

Aaron Guerra

Discussion Questions

1. What signs should you look for to know that your codebook is ready and doesn't need more editing?

I think that you can know a codebook is ready in a similar way to knowing when a project is ready - you test it with someone else and they are able to use it successfully, ideally someone who is not a domain expert in what the coding is for. In the creation of the codebook you (or you and your team) are deep in the weeds and are more confident than someone who would be new to the codebook. Testing it on a non-domain expert would show that the codebook is adequately detailed, covers the breadth of topics needed, and is understandable and repeatable. Having a high intercoder reliability between the author and tester would demonstrate an excellent codebook.

2. Beyond the best practices already discussed, what other best practices would make sense to employ in using quantitative manual content analysis?

I think best practices not already addressed cover teamwork throughout the process. Cooperatively producing the codebook with co-authors or other stakeholders when possible is a best practice due to the diversity of opinions and perspectives that can contribute to how different individuals might attempt to code the same idea. Similarly, co-production of the codebook would resolve some initial disagreements about how to code certain items. Other best practices could include different versions of codebooks that are used on the same dataset to see how differences in the codebook could produce codes with more or less agreement among coders. Finally, some sort of mixed-methods approach, such as deep reading or qualitative coding of certain documents could provide deeper insights into disagreements in codes or other codes that would be useful.

Project Questions

1. Work with your partner or team to develop a single variable of interest and a brief codebook, using a corpus of your choice. What was the process like? What were the easy parts, and what were the challenges?

We selected the corpus of California legislative bills, and our initial codebook was as follows:

CODEBOOK

Procedure: Examine the legislative digest first as that often contains a simple summary of the bill text and changes

Latent Variable of Interest: symbolic versus substantive bill content

Symbolic (0)

- not changing any existing policy
- delegating / emphasizing power
- most likely have no appropriation, fiscal committee, or local program present (found at end of legislative digest)

example text

- “prohibit…from making any rules”
- “relative to”

Substantive (1)

- change existing policy
- change in numbers/money/etc.
- creating new policy
- extreme detail

example text

- “an act to amend…”
- “an act to add…”
- “tax levy”

While I wasn't specifically involved in choosing the first variable of interest and codebook (due to my absence for MPSA), I can speak to this process for the second variable. I found that it is fairly simple to come up with a variable of interest but deciding on how to strictly define that idea is more difficult. One of the biggest difficulties is crosswalking between what I intuitively think of for defining two codes and how you practically implement that idea given the limitations of the text. For example, I think of state bills for local requirements more in terms of changes to individuals, such as changes to healthcare or tax law or similar, but after comparing this with the corpus they primarily relate to a federalist balance of power between state and local and constitutional requirements for the state to fund programs it mandates locally.

2. Subset your corpus to 20 observations, and have each partner code those 20 items independently. What was the process like? How confident did you feel in your coding, and what did you observe from spending time with the text?

```r
library(tidyverse)

set.seed(1121) # seed for replication for initial coding

# Load data
df <- readRDS("hw1_data.rds")

df_subset <- df[sample(nrow(df), 20), ] %>%
  mutate(text = str_squish(text))

codes <- c(0,
           1,
           1,
           0,
           1,
           1,
           0,
           1,
           1,
           1,
           1,
           1,
           0,
           0,
           0,
           0,
           1,
           1,
           1,
            1)

df_final <- df_subset %>%
  mutate(aaron_codes = codes)

write.csv(df_final, "aaron_codes.csv", row.names=TRUE)
```

The process of coding was definitely difficult, although I think it would have been harder in a categorical coding versus binary coding. One of the biggest difficulties of this coding was the length of each bill meant that there were different components which could fall in either substantive or symbolic categories, and determining which was bigger was difficult. I found myself relying on some heuristics (which were included in the codebook), particularly the appropriation/fiscal committee/local program to define when I was unsure.

3. Calculate your inter-coder reliability (follow Professor Freelon's instructions about how to format the .csv file!). What factors help explain how high or low it is? Which particular observations did you code differently, and how might you update the coding process to improve the inter-coder reliability?

Our intercoder reliability (with five coders) was 0.434. In 10 cases we agreed on all cases, in 4 cases there was one person who disagreed (in 3 of those it was the same person), and in the last 6 cases there was significant disagreement on the coding. One explanation for the low reliability might be in the fact that our coders were added at different stages, where two people were present and discussed in-person the differences between substantive and symbolic, but three of us were exposed to this coding process without participating in its creation. I think this represents a issue in the literature - it's not explicitly addressed in any of these readings, but some protocols have senior/experienced researchers create the codebook which is used by the coders, while some have it as a collaborative process. This experience suggests to me that the collaborative process would be potentially more helpful as any individual biases or perspectives can be addressed rather than latent in one's coding procedure.

4. Now repeat steps 1-3 again, but using a different variable of interest OR a different set of coding rules for your same variable of interest.

```
set.seed(1303) # seed for replication for initial coding

df_subset2 <- df[sample(nrow(df), 20), ] %>%
  mutate(text = str_squish(text))

codes2 <- c(0,
            2,
           0,
           2,
           0,
           0,
           0,
           0,
           1,
           1,
           0,
           0,
           1,
           1,
           0,
           2,
           1,
           2,
           2,
           2)

df_final2 <- df_subset2 %>%
  mutate(aaron_codes = codes2)

write.csv(df_final2, "aaron_codes2.csv", row.names=TRUE)
```

In our second pass through, we decided to instead study state vs local policy, and we created a codebook as follows:

CODEBOOK

Procedure: Examine the legislative digest first as that often contains a simple summary of the bill text and changes

Latent Variable of Interest: state requirements versus local requirements in California bills

- *State (0)*
  requires action of a department, agency, or individual employed by the State
- "would require the […] department to"
- Appropriation likely indicates bureaucratic

Local (1)

- Requires action of anyone not employed by or working for the State, including local governments, private businesses, and individual citizens.
- "This bill would authorize any local agency"

- “require the above-described lifeguards”
- Local program likely indicates public-facing
- “Tax levy”

Both (2) - Meets requirements of both of the above.

Coding under this scheme felt more confident for me, although given that I produced most of the codebook this would make sense - I have spent more time with the codes and would feel more confident in which category to assign documents to. Also, it was not any harder to code in this tri-categorical scheme than in the previous binary scheme, although I still believe that having more codes would be more difficult.

Unfortunately the inter-coder reliability for this round was even lower than the first: a Krippendorff’s Alpha of 0.385. Looking at the codes, there is one coder who seems to have a different understanding of the codes - in 7 of 20 codes, they disagree with the other four coders. This represents some fundamental misunderstanding between coders and would suggest a collaborative discussion over the codes would be useful to revise the codebook. In three cases, this code identifies documents that all other coders identify as “state” as either local or both. This suggests that revising this distincition would be useful in improving agreement among coders.

5. Finally, discuss: what worked and what didn’t, and why? Explain the similarities and differences between your two coding approaches and what you learned.

The biggest lesson is that coding with a larger team (5 coders) without any in-person or live discussion is a difficult process and leads to significant disagreement and misunderstandings of the codebook. In the first coding approach we used more jargon-y terms (substantive and symbolic) with mixed results, and in the second we used what I thought to be more simple terms (state and local) yet had worse results - counterintuitively, as political scientists we may have a better collective understanding of substnative/symbolic compared to state/local. Alternatively, something in the coding scheme for substantive/symbolic was more clear and agreeable than in the state/local coding scheme, although further discussions would be needed to elucidate what. Another explanation could be the inclusion of a third code in the state/local (coding for both), which could introduce uncertainty.