Table Scraps Supplemental Materials 2

April 29, 2020

1 Establishing Saturation

We establish saturation in our codeset by monitoring the number of unique codes with respect to the number of repos included in our technical observation study. Approaching 50 repos we notice the size of the codeset leveling off. At this point, we determine the codeset has reached saturation, adequately describing data wrangling actions and processes in this domain.

[37]:

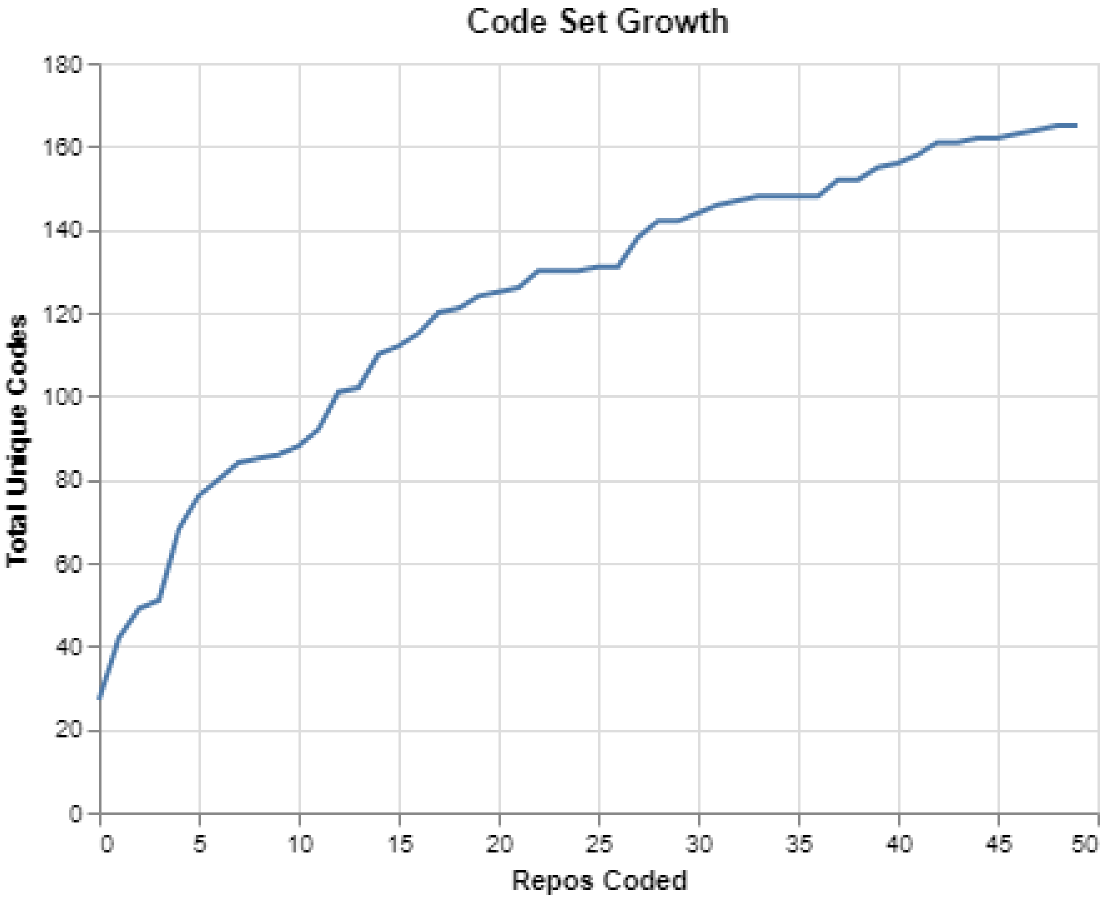


Fig. S1: Codeset growth per repo coded. For each notebook included in analysis list which codes were introduced to the code set. After 23 notebooks, some computational notebooks didn't add any new codes. By 50 notebooks, code set growth was so minimal that we declared our code set converged.

1.1 Newly introduced codes by repo

Below we explicitly list which repos defined new open codes in our codeset. Repos are ordered by when they were coded in our technical observation study.

1. **2019-04-democratic-candidate-codonors**: create child table, trim by categorical value, repetitive code, count unique values, figure a rate, create annotations, outer join, compare groups, deduplicate, create soft key, group by variable, create a frequency table, trim by quantitative threshold, gather, load, format values, export, canonicalize variable names, remove variables, peek at data, govt data portal, union datasets, sort, change var type, self join dataset, aggregate, standardize categorical variables, construct a subroutine, align variables
2. **california-ccscore-analysis**: describe statistically, show trend over time, remove incomplete data, visualize data, calculate spread, count number of rows, standardize variable, trim by date range, inspect data schema, identify extreme values, cross tabulate, divide & conquer, calculate change over time, trim fat
3. **california-crop-production-wages-analysis**: adjust for inflation, construct data manually, construct data pipeline, wrangle data for graphics, combine periodic data, trim by geographic area, inner join, lookup table values
4. **census-hard-to-map-analysis**: parse variable, tolerate dirty data
5. **long-term-care-db**: generate high-level summary, generate dataset identification, scrape web for data, create lookup table, refine table, fill in na values after an outer join, count the data, combine categorical values, edit values, replace na values, use non-public, provided data
6. **2018-voter-registration**: impute missing data, calculate a statistic, aggregate join, join aggregate, extract data from pdf, assign ranks
7. **heat-index**: generate data computationally, cartesian product, examine relationship, compute index number
8. **2016-11-bellwether-counties**: rolling window calculation, get extreme values, create a unique key, remove non-data rows, spread table, use academic data
9. **2018-05-31-crime-and-heat-analysis**: split, compute, and merge, merge seemingly disparate datasets
10. **2016-09-shy-trumpers**: use another news orgs data
11. **the-cube-root-law**: domain-specific performance metric, use public data
12. **2016-04-republican-donor-movements**: explore dynamic network flow
13. **california-h2a-visas-analysis**: consolidate variables, temporary joining column, preserve existing values, select rows with missing values, resolve entities, api request, schema drift
14. **Endangered-Species-Act-Louisiana**: scale values

15. **Power_of_Irma**: variable replacement, set data confidence threshold, use data from colleague, fix incorrect calculation, create togglable operations, use previously cleaned data, interpret statistical/ml model

16. **wikipedia-rankings**: collect raw data, explain variance

17. **babyname_politics**: resort after merge, data loss from aggregation

18. **2015-11-refugees-in-the-united-states**: test for equality, make an incorrect conclusion, lossy join

19. **employment-discrimination**: replace variable levels

20. **bechdel**: data type shyness

21. **bob-ross**:

22. **nyc-trips**: full join

23. **work-from-home**: concat parallel datasets, create a flag, copy table schema, data too large for repo, split and compute

24. **buster-posey-mvp**:

25. **verge-uber-launch-dates**:

26. **vox-central-line-infections**: geolocate dataset records, report rows with column number discrepancies

27. **prison-admissions**:

28. **school-star-ratings-2018**: remove duplicate variables

29. **electric-car-charging-points**: perform network analysis

30. **internal-migration-london**:

31. **midwife-led-units**: freedom of information data

32. **librarians**:

33. **infrastructure-jobs**:

34. **federal_employees_trump_2017**:

35. **2019-ems-analysis**:

36. **auditData**:

37. **lending-club**:

38. **new-york-schools-assessment**:

39. **skatemusic**:

40. **awb-notebook**: test for null values, silently dropping values after groupby

41. **201901-hospitalquality**:

42. **general-election-2015-classification-tree**: wrangle data for model, check for nas

43. **201901-achievementgap**: bin values, query database
44. **school-choice**: transpose
45. **1805-regionen im fokus des US-praesidenten**:
46. **swana-population-map**:
47. **california-buildings-in-severe-fire-hazard-zones**: search for clusters
48. **us-weather-history**: validate data quality with domain-specific rules
49. **gunsales**: adjust for season
50. **demolitions**:

2 Incorporating diversity

In order to prevent this code set from being biased by one individual or organization's data wrangling behavior, we deliberately sought out notebooks from a variety of news organizations and data journalists. This analysis comes from, but is not limited to, news organizations that constitute ``major players'' in data journalism.

2.1 Prolificness of news organizations

Some news organizations are more engaged in data journalism than others. In order for the result of our technical observation study to be representative of the practices of a variety of organizations, we deliberately selected notebooks for inclusion in our technical observation study by news organizations across the spectrum of prolificness in this genre of journalism.

We ranked these organizations by two metrics based on our pool of journalistic code repositories containing data analysis:

- The count of individual code repositories
- The number of commits by journalists working for different news organizations

2.1.1 By number of repos

Most news organizations, including *BuzzFeed News*, *Los Angeles Times*, and the *Austin American-Statesman*, create one repository per analysis work flow. We include at least one repository from the top 19 news organizations by the number of unique repositories in our pool journalistic code repositories containing data analysis. We also deliberately select repositories from news organization that only have one repository in this pool.

[20]:

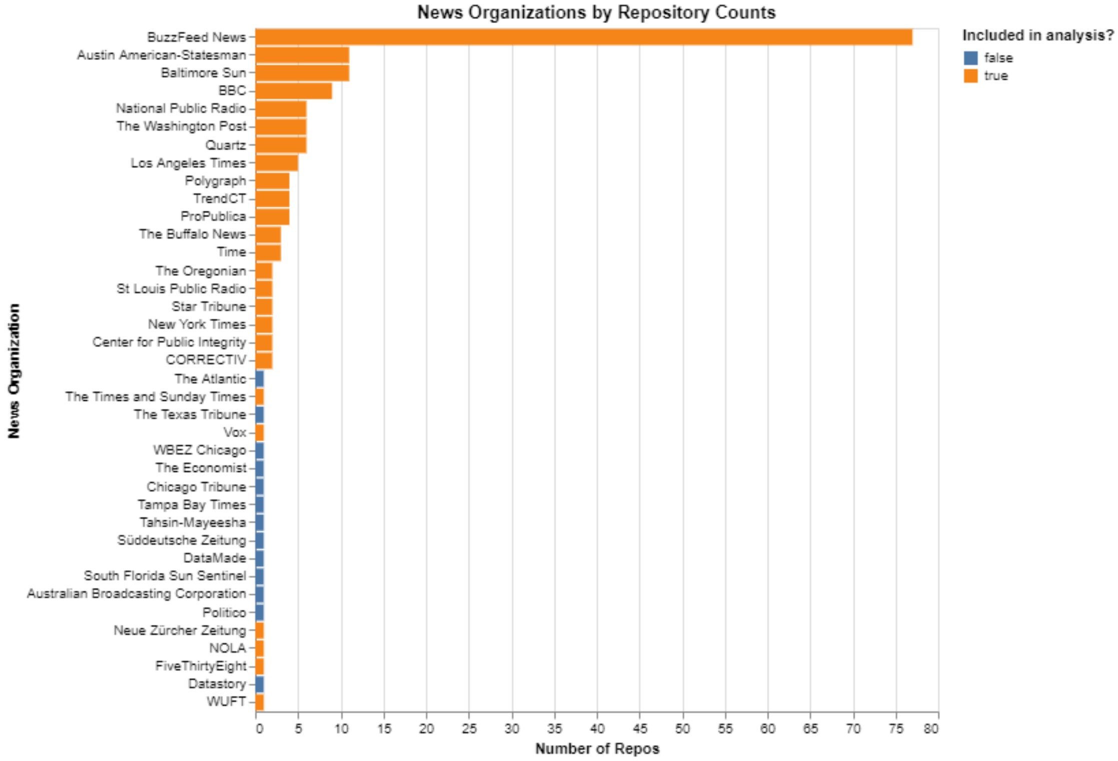


Fig. S2: Repository count by news organization. This bar chart show the number of repositories per news organization in our curated pool of journalistic, data-analysis repositories, color-coded by whether at least one repository from that news organization was included in our technical observation study. Orange values indicate the news organization was included and blue indicates otherwise.

2.1.2 By commits

However, one limitation of ranking news organizations by the number of repositories that some organizations, such as *FiveThirtyEight* keep computational notebooks for multiple data journalism articles in one master code repository. A *commit* in Git can be thought of as a unit of change. Thus, the more a repository has changed overtime, the more commits. If a news organization is only using one repository for all their data journalism work, then it should have lots of commits.

When ranking news organizations by commit counts, our qualitative analysis includes include the top 18 news organizations by commit count in addition to news organizations with only a few commits in our pool of journalistic code repositories containing data analysis.

[12]:

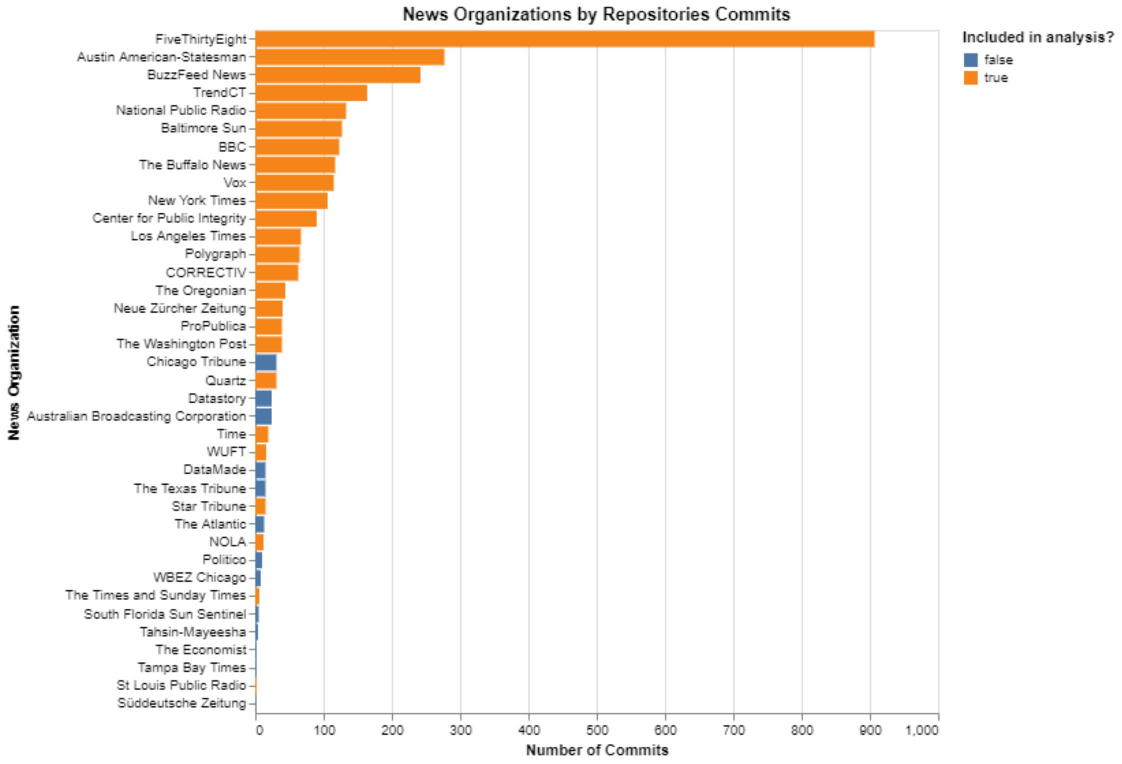


Fig. S3: News organizations ranked by number of commits. This bar chart show the number of commits per users associated with various news organization in our curated pool of journalistic, data-analysis repositories. The chart is color-coded by whether at least one repository from that news organization was included in our technical observation study. Orange values indicate the news organization was included and blue indicates otherwise.

This analysis includes 25 news organizations out of 37 that had computational notebooks deemed relevant to this analysis (67.57%).

Organization	Is included?
Austin American-Statesman	Yes
Australian Broadcasting Corporation	No
BBC	Yes
Baltimore Sun	Yes
BuzzFeed News	Yes
CORRECTIV	Yes
Center for Public Integrity	Yes
Chicago Tribune	No
DataMade	No
Datastory	No
FiveThirtyEight	Yes
Los Angeles Times	Yes
NOLA	Yes
National Public Radio	Yes
Neue Zürcher Zeitung	Yes

Organization	Is included?
New York Times	Yes
Politico	No
Polygraph	Yes
ProPublica	Yes
Quartz	Yes
South Florida Sun Sentinel	No
St Louis Public Radio	Yes
Star Tribune	Yes
Süddeutsche Zeitung	No
Tampa Bay Times	No
The Atlantic	No
The Buffalo News	Yes
The Economist	No
The Oregonian	Yes
The Texas Tribune	No
The Times and Sunday Times	Yes
The Washington Post	Yes
Time	Yes
TrendCT	Yes
Vox	Yes
WBEZ Chicago	No
WUFT	Yes

2.2 Prolificness of individual journalists

In addition to taking steps to incorporate comprehensiveness and diversity of news organization into our descriptive taxonomy, we also attempt to add comprehensiveness and diversity in the individual journalists.

We exclude some data journalist with commits from this summary because their commits were insignificant contributions to repos such as comments, README file updates, initial repo setup, and general code clean up.

- Andrei Scheinkman, *FiveThirtyEight*
- Dhrumil Mehta, *FiveThirtyEight*
- Stephen Turner, *FiveThirtyEight*
- Nate Silver, *FiveThirtyEight*
- Dan Nguyen, *The Upshot*
- Derek Willis, *BuzzFeed News*

Note that this summary also excludes journalists who:

- Worked collaboratively and only one of them committed code.
  - Matt Stevens

    - Adam Pearce
- Only were included in the technical observations study via Observable notebooks
    - Sahil Chinoy
- Did not commit their own code. For example, *FiveThirtyEight* code appears to be committed by someone else.
    - Rob Arthur
    - Stefano Ceccon
    - Walt Hickey

2.2.1 By commits

[25]:

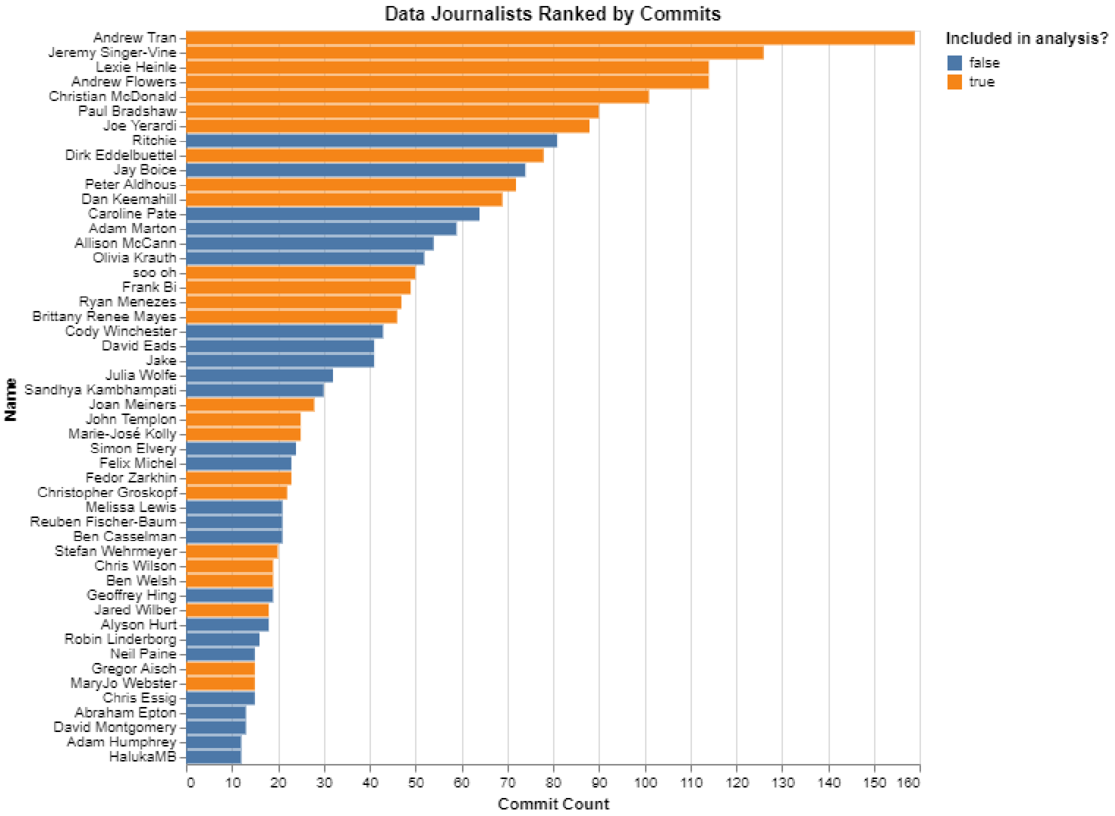


Fig. S4: Data journalists who authored code repositories in our pool of journalistic, data-analysis repos, ranked by number of commits. This chart is color-coded orange to indicate that the individual authored an analysis included in our technical observation study.

2.2.2 By followers

Our qualitative analysis is based on repositories authored by the top eight data journalists ranked by the number of followers in addition to many GitHub users with less followers.

[18]:

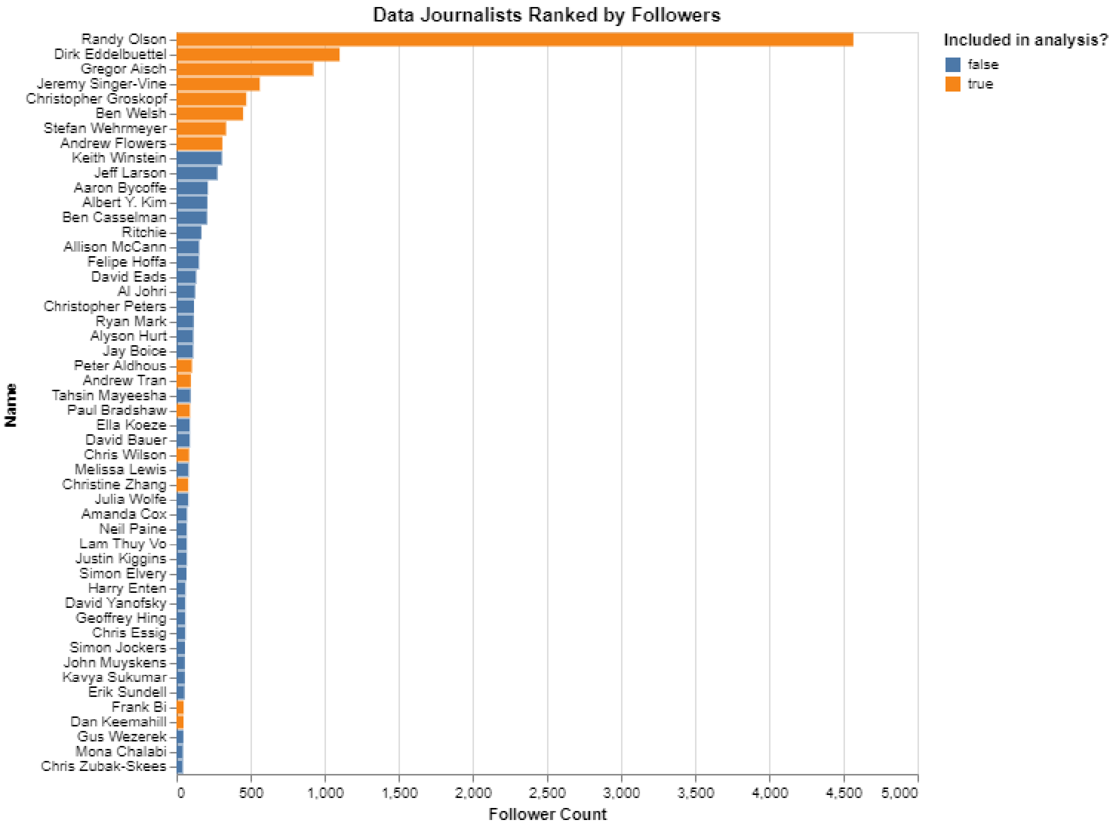
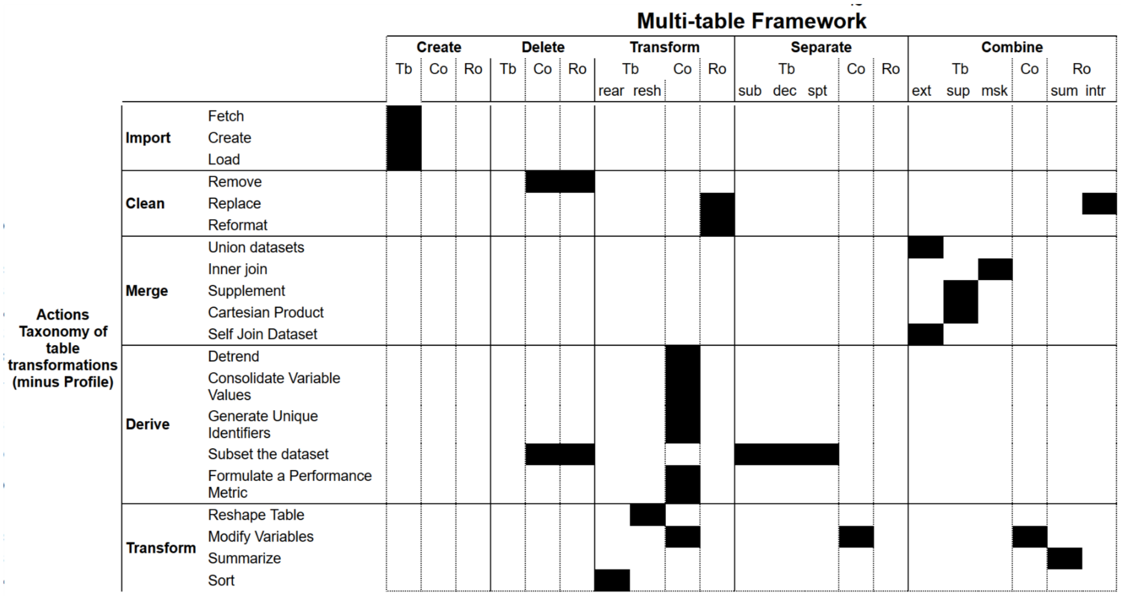


Figure S5: Data journalists who authored code repositories in our pool of journalistic, data-analysis repos, ranked by number of followers on GitHub. This chart is color-coded orange to indicate that the individual authored an analysis included in our technical observation study.

3 Descriptive cross-check of multi-table framework

We cross check the descriptive power of our multi-table framework for data wrangling by comparing against the high-level axial codes in our descriptive action taxonomy. We only include actions codes that correspond with table operations, hence excluding codes in the Profile branch.



Multi-table Framework

			Create			Delete			Transform				Separate					Combine					
			Tb	Co	Ro	Tb	Co	Ro	Tb		Co	Ro	Tb			Co	Ro	Tb			Co	Ro	
									rear	resh			sub	dec	spt			ext	sup	msk		sum	intr
Actions Taxonomy of table transformations (minus Profile)	Import	Fetch	■																				
		Create	■																				
		Load	■																				
	Clean	Remove					■	■															
		Replace										■											■
		Reformat										■											
	Merge	Union datasets																■					
		Inner join																		■			
		Supplement																	■				
		Cartesian Product																	■				
		Self Join Dataset																■					
	Derive	Detrend									■												
		Consolidate Variable Values									■												
		Generate Unique Identifiers									■												
		Subset the dataset					■	■					■	■	■								
		Formulate a Performance Metric									■												
	Transform	Reshape Table								■													
		Modify Variables									■					■					■		
		Summarize																				■	
		Sort							■														