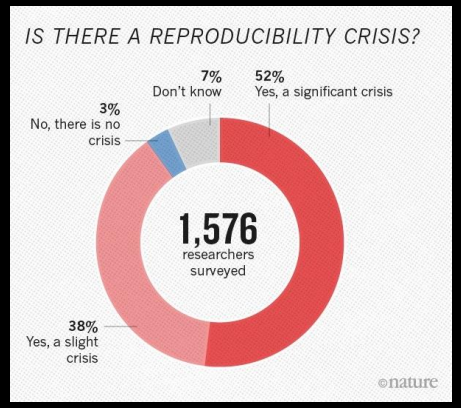
THE DATAGOOD(R) PACKAGE: AN OPEN SCIENCE APPROACH TO DATA ENGINEERING

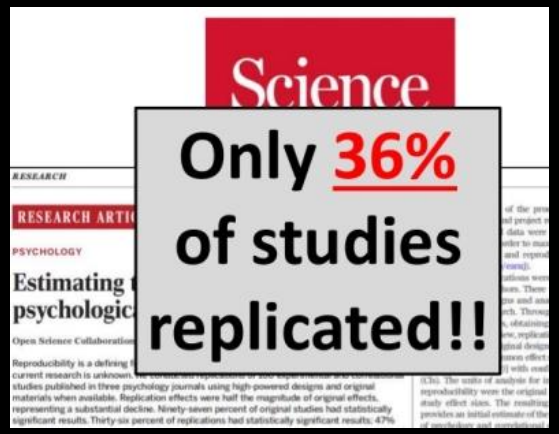
Jesse Lecy

Associate Professor ::: Public Affairs (**ASU**)

Data Scientist ::: The Urban Institute

The Reproducibility Crisis in Science





Sources of Replication Failure

- Fraud
- Editorial / Reviewer Discretion
- P-value Hacking

incentive problems

- Study Sample Outside of CI
- Errors in Statistical Methodology
- Errors in Computing

engineering problems

IMPROVING REPRODUCIBILITY THROUGH DATA PROVENANCE

DATA PROVENANCE:

Ability to the steps in the data engineering process to reproduce the same **research dataset** from the original sources

FAIR Data Standards Extend Provenance:

- FAIR (**Findable – Accessible – Interoperable – Reusable**)
- Platforming the data so that others can replicate your results or extend the project
- Enable data sharing when original data has privacy protections

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., ... & Mons, B. (2016). The FAIR Guiding Principles for scientific data management and stewardship. Scientific data, 3(1), 1-9.

There is an urgent need to improve the infrastructure supporting the reuse of scholarly data. A diverse set of stakeholders—representing academia, industry, funding agencies, and scholarly publishers—have come together to design and jointly endorse a concise and measurable set of principles that we refer to as the FAIR Data Principles. The intent is that these may act as a guideline for those wishing to enhance the reusability of their data holdings. Distinct from peer initiatives that focus on the human scholar, the FAIR Principles put specific emphasis on enhancing the ability of machines to automatically find and use the data, in addition to supporting its reuse by individuals.

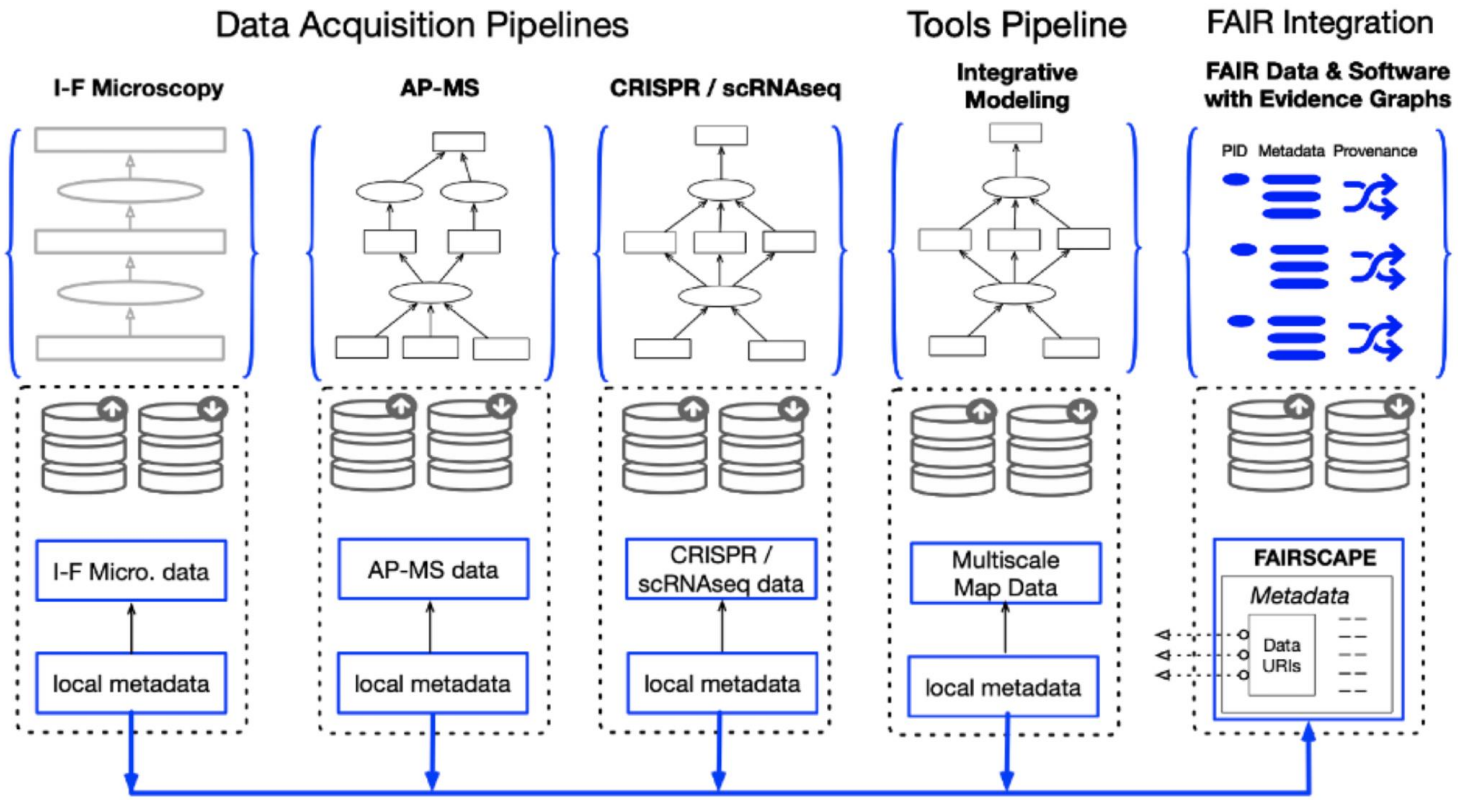
https://cm4ai.org/standards/
https://commonfund.nih.gov/bridge2ai/news

Data Provenance Requires Explicit:

- data engineering steps
- data acquisition steps
- data versioning

```
EXAMPLE FAIR DATA CONFIG FILE:

run_metadata:
  local_data_registry_url: https://localhost:8000/api/
  remote_data_registry_url: https://fairdatapipeline.org/api/
  script: |-
    R -f submission_script.R

read:
- data_product: records/SARS-CoV-2/cases-and-management
    version: 0.20210414.0
    namespace: soniamitchell

write:
- data_product: records/SARS-CoV-2/ambulance
  description: Ambulance data
  use:
    version: 0.20210414.0
- data_product: records/SARS-CoV-2/calls
  description: Calls data
  use:
    version: 0.20210414.0
```

EXAMPLE: The Cell Maps for AI (CM4AI) Standards Module



The Cell Maps for AI (CM4AI) Standards Module will provide original, interim and final datasets and software from the CM4AI Data Acquisition and Tools pipeline, with final AI-ready results, as comprehensively FAIR (Findable – Accessible – Interoperable – Reusable) digital objects for uptake and reuse by biomedical Artificial Intelligence (AI) applications. These objects will be provided within a computational digital commons environment based on the FAIRSCAPE framework.

COMPUTING CHALLENGES:
MANAGING COMPLEXITY

From: Gentzkow, M., & Shapiro, J. M. (2014). Code and data for the social sciences: A practitioner's guide. Chicago, IL: University of Chicago.

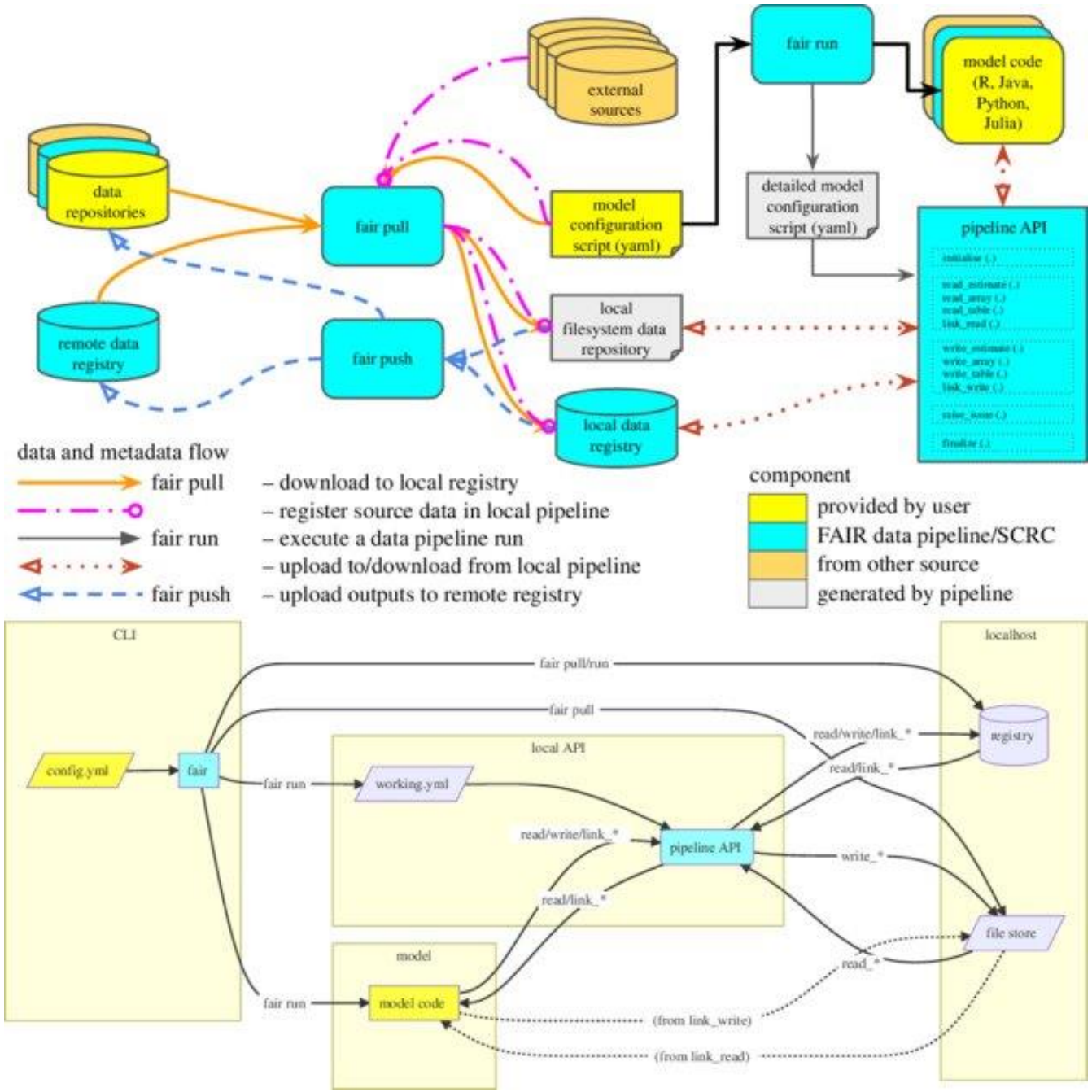
> **Though we all write code for a living**, few of the economists, political scientists, psychologists, sociologists, or other empirical researchers we know **HAVE ANY FORMAL TRAINING IN COMPUTER SCIENCE**. Most of them picked up the basics of programming without much effort and have never given it much thought since. Saying they should spend more time thinking about the way they write code would be like telling a novelist that she should spend more time thinking about how best to use Microsoft Word.

From: Gentzkow, M., & Shapiro, J. M. (2014). Code and data for the social sciences: A practitioner's guide. Chicago, IL: University of Chicago.

> Here is a good rule of thumb: If you are trying to solve a problem, and there are multi-billion-dollar firms whose entire business model depends on solving the same problem, and there are whole courses at your university devoted to how to solve that problem, you might want to figure out what the experts do and see if you can't learn something from it.

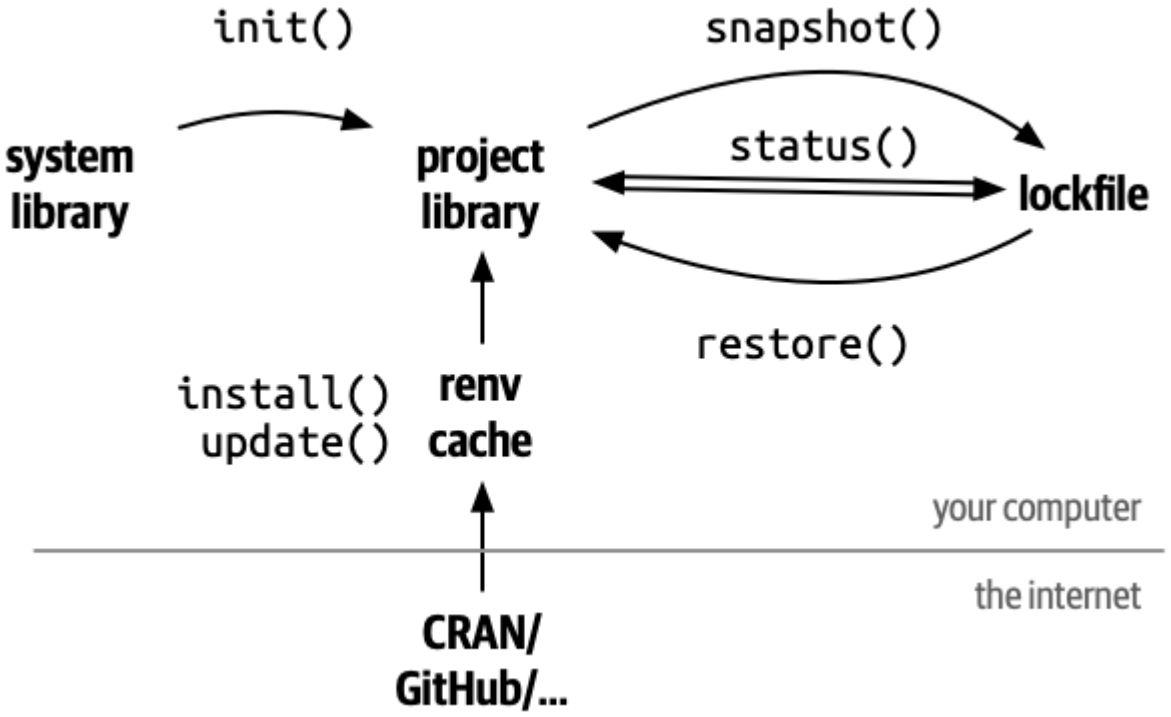
The Unbearable Complexity of Depending

```
GEM
  remote: https://rubygems.org/
  specs:
    activesupport (7.0.8)
      concurrent-ruby (~> 1.0, >= 1.0.2)
      i18n (>= 1.6, < 2)
      minitest (>= 5.1)
      tzinfo (~> 2.0)
    addressable (2.8.4)
      public_suffix (>= 2.0.2, < 6.0)
    coffee-script (2.4.1)
      coffee-script-source
      execjs
    coffee-script-source (1.11.1)
    colorator (1.1.0)
    commonmarker (0.23.10)
    concurrent-ruby (1.2.2)
    dnsruby (1.70.0)
      simpleidn (~> 0.2.1)
    em-websocket (0.5.3)
      eventmachine (>= 0.12.9)
      http_parser.rb (~> 0)
    ethon (0.16.0)
      ffi (>= 1.15.0)
    eventmachine (1.2.7)
    execjs (2.8.1)
    faraday (2.7.5)
      faraday-net_http (>= 2.0, < 3.1)
      ruby2_keywords (>= 0.0.4)
    faraday-net_http (3.0.2)
    ffi (1.15.5)
    forwardable-extended (2.6.0)
    gemoji (3.0.1)
    github-pages (228)
      github-pages-health-check (= 1.17.9)
      jekyll (= 3.9.3)
      jekyll-avatar (= 0.7.0)
      jekyll-coffeescript (= 1.1.1)
      jekyll-commonmark-ghpages (= 0.4.0)
      jekyll-default-layout (= 0.1.4)
      jekyll-feed (= 0.15.1)
      jekyll-gist (= 1.5.0)
      jekyll-github-metadata (= 2.13.0)
      jekyll-include-cache (= 0.2.1)
      jekyll-mentions (= 1.6.0)
      jekyll-optional-front-matter (= 0.3.2)
      jekyll-paginate (= 1.1.0)
      jekyll-readme-index (= 0.3.0)
      jekyll-redirect-from (= 0.16.0)
      jekyll-relative-links (= 0.6.1)
      jekyll-remote-theme (= 0.4.3)
      jekyll-sass-converter (= 1.5.2)
      jekyll-seo-tag (= 2.8.0)
      jekyll-sitemap (= 1.4.0)
      jekyll-swiss (= 1.0.0)
      jekyll-theme-architect (= 0.2.0)
      jekyll-theme-cayman (= 0.2.0)
      jekyll-theme-dinky (= 0.2.0)
      jekyll-theme-hacker (= 0.2.0)
      jekyll-theme-leap-day (= 0.2.0)
      jekyll-theme-merlot (= 0.2.0)
      jekyll-theme-midnight (= 0.2.0)
      jekyll-theme-minimal (= 0.2.0)
      jekyll-theme-modernist (= 0.2.0)
      jekyll-theme-primer (= 0.6.0)
      jekyll-theme-slate (= 0.2.0)
      jekyll-theme-tactile (= 0.2.0)
      jekyll-theme-time-machine (= 0.2.0)
      jekyll-titles-from-headings (= 0.5.3)
      jemoji (= 0.12.0)
      kramdown (= 2.3.2)
      kramdown-parser-gfm (= 1.1.0)
      liquid (= 4.0.4)
      mercenary (~> 0.3)
      minima (= 2.5.1)
      nokogiri (>= 1.13.6, < 2.0)
      rouge (= 3.26.0)
      terminal-table (~> 1.4)
    github-pages-health-check (1.17.9)
      addressable (~> 2.3)
      dnsruby (~> 1.60)
      octokit (~> 4.0)
      public_suffix (>= 3.0, < 5.0)
      typhoeus (~> 1.3)
    html-pipeline (2.14.3)
      activesupport (>= 2)
      nokogiri (>= 1.4)
    http_parser.rb (0.8.0)
    i18n (1.14.1)
      concurrent-ruby (~> 1.0)
    jekyll (3.9.3)
      addressable (~> 2.4)
      colorator (~> 1.0)
      em-websocket (~> 0.5)
      i18n (>= 0.7, < 2)
      jekyll-sass-converter (~> 1.0)
      jekyll-watch (~> 2.0)
      kramdown (>= 1.17, < 3)
      liquid (~> 4.0)
      mercenary (~> 0.3.3)
      pathutil (~> 0.9)
      rouge (>= 1.7, < 4)
      safe_yaml (~> 1.0)
    jekyll-avatar (0.7.0)
      jekyll (>= 3.0, < 5.0)
    jekyll-coffeescript (1.1.1)
      coffee-script (~> 2.2)
      coffee-script-source (~> 1.11.1)
    jekyll-commonmark (1.4.0)
      commonmarker (~> 0.22)
    jekyll-commonmark-ghpages (0.4.0)
      commonmarker (~> 0.23.7)
      jekyll (~> 3.9.0)
      jekyll-commonmark (~> 1.4.0)
      rouge (>= 2.0, < 5.0)
    jekyll-default-layout (0.1.4)
      jekyll (~> 3.0)
    jekyll-feed (0.15.1)
      jekyll (>= 3.7, < 5.0)
    jekyll-gist (1.5.0)
      octokit (~> 4.2)
    jekyll-github-metadata (2.13.0)
      jekyll (>= 3.4, < 5.0)
      octokit (~> 4.0, != 4.4.0)
    jekyll-include-cache ( (>= 1.0, <= 3.0.0, != 2.0.0)
    rubyzip (>= 1.3.0, < 3.0)
    jekyll-sass-converter (1.5.2)
      sass (~> 3.4)
    jekyll-seo-tag (2.8.0)
      jekyll (>= 3.8, < 5.0)
    jekyll-sitemap (1.4.0)
      jekyll (>= 3.7, < 5.0)
    jekyll-swiss (1.0.0)
    jekyll-theme-architect (0.2.0)
      jekyll (> 3.5, < 5.0)
      jekyll-seo-tag (~> 2.0)
    jekyll-theme-cayman (0.2.0)
      jekyll (> 3.5, < 5.0)
      jekyll-seo-tag (~> 2.0)
    jekyll-theme-dinky (0.2.0)
      jekyll (> 3.5, < 5.0)
      jekyll-seo-tag (~> 2.0)
    jekyll-theme-hacker (0.2.0)
      jekyll (> 3.5, < 5.0)
      jekyll-seo-tag (~> 2.0)
    jekyll-theme-leap-day (0.2.0)
      jekyll (> 3.5, < 5.0)
      jekyll-seo-tag (~> 2.0)
    jekyll-theme-merlot (0.2.0)
      jekyll (> 3.5, < 5.0)
      jekyll-seo-tag (~> 2.0)
    jekyll-theme-midnight (0.2.0)
      jekyll (> 3.5, < 5.0)
      jekyll-seo-tag (~> 2.0)
    jekyll-theme-minimal (0.2.0)
      jekyll (> 3.5, < 5.0)
      jekyll-seo-tag (~> 2.0)
    jekyll-theme-modernist (0.2.0)
      jekyll (> 3.5, < 5.0)
      jekyll-seo-tag (~> 2.0)
    jekyll-theme-primer (0.6.0)
      jekyll (> 3.5, < 5.0)
      jekyll-github-metadata (~> 2.9)
      jekyll-seo-tag (~> 2.0)
    jekyll-theme-slate (0.2.0)
      jekyll (> 3.5, < 5.0)
      jekyll-seo-tag (~> 2.0)
    jekyll-theme-tactile (0.2.0)
      jekyll (> 3.5, < 5.0)
      jekyll-seo-tag (~> 2.0)
    jekyll-theme-time-machine (0.2.0)
      jekyll (> 3.5, < 5.0)
      jekyll-seo-tag (~> 2.0)
    jekyll-titles-from-headings (0.5.3)
      jekyll (>= 3.3, < 5.0)
    jekyll-watch (2.2.1)
      listen (~> 3.0)
    jemoji (0.12.0)
      gemoji (~> 3.0)
      html-pipeline (~> 2.2)
      jekyll (>= 3.0, < 5.0)
    kramdown (2.3.2)
      rexml
    kramdown-parser-gfm (1.1.0)
      kramdown (~> 2.0)
    liquid (4.0.4)
    listen (3.8.0)
      rb-fsevent (~> 0.10, >= 0.10.3)
      rb-inotify (~> 0.9, >= 0.9.10)
    mercenary (0.3.6)
    minima (2.5.1)
      jekyll (>= 3.5, < 5.0)
      jekyll-feed (~> 0.9)
      jekyll-seo-tag (~> 2.1)
    minitest (5.20.0)
    nokogiri (1.15.2-arm64-darwin)
      racc (~> 1.4)
    nokogiri (1.15.2-x86_64-linux)
      racc (~> 1.4)
    octokit (4.25.1)
      faraday (>= 1, < 3)
      sawyer (~> 0.9)
    pathutil (0.16.2)
      forwardable-extended (~> 2.6)
    public_suffix (4.0.7)
    racc (1.6.2)
    rb-fsevent (0.11.2)
    rb-inotify (0.10.1)
      ffi (~> 1.0)
    rexml (3.2.5)
    rouge (3.26.0)
    ruby2_keywords (0.0.5)
    rubyzip (2.3.2)
    safe_yaml (1.0.5)
    sass (3.7.4)
      sass-listen (~> 4.0.0)
    sass-listen (4.0.0)
      rb-fsevent (~> 0.9, >= 0.9.4)
      rb-inotify (~> 0.9, >= 0.9.7)
    sawyer (0.9.2)
      addressable (>= 2.3.5)
      faraday (>= 0.17.3, < 3)
    simpleidn (0.2.1)
      unf (~> 0.1.4)
    terminal-table (1.8.0)
      unicode-display_width (~> 1.1, >= 1.1.1)
    typhoeus (1.4.0)
      ethon (>= 0.9.0)
    tzinfo (2.0.6)
      concurrent-ruby (~> 1.0)
    unf (0.1.4)
      unf_ext
    unf_ext (0.0.8.2)
    unicode-display_width (1.8.0)

PLATFORMS
  arm64-darwin-21
  x86_64-linux

DEPENDENCIES
  github-pages
  http_parser.rb (~> 0.6.0)
  jekyll-feed (~> 0.12)
  tzinfo (>= 1, < 3)
  tzinfo-data
  wdm (~> 0.1.1)

BUNDLED WITH
   2.3.16

0.2.1)
      jekyll (>= 3.7, < 5.0)
    jekyll-mentions (1.6.0)
      html-pipeline (~> 2.3)
      jekyll (>= 3.7, < 5.0)
    jekyll-optional-front-matter (0.3.2)
      jekyll (>= 3.0, < 5.0)
    jekyll-paginate (1.1.0)
    jekyll-readme-index (0.3.0)
      jekyll (>= 3.0, < 5.0)
    jekyll-redirect-from (0.16.0)
      jekyll (>= 3.3, < 5.0)
    jekyll-relative-links (0.6.1)
      jekyll (>= 3.3, < 5.0)
    jekyll-remote-theme (0.4.3)
      addressable (~> 2.0)
      jekyll (>= 3.5, < 5.0)
      jekyll-sass-converter
```



FAIR Data pipelines

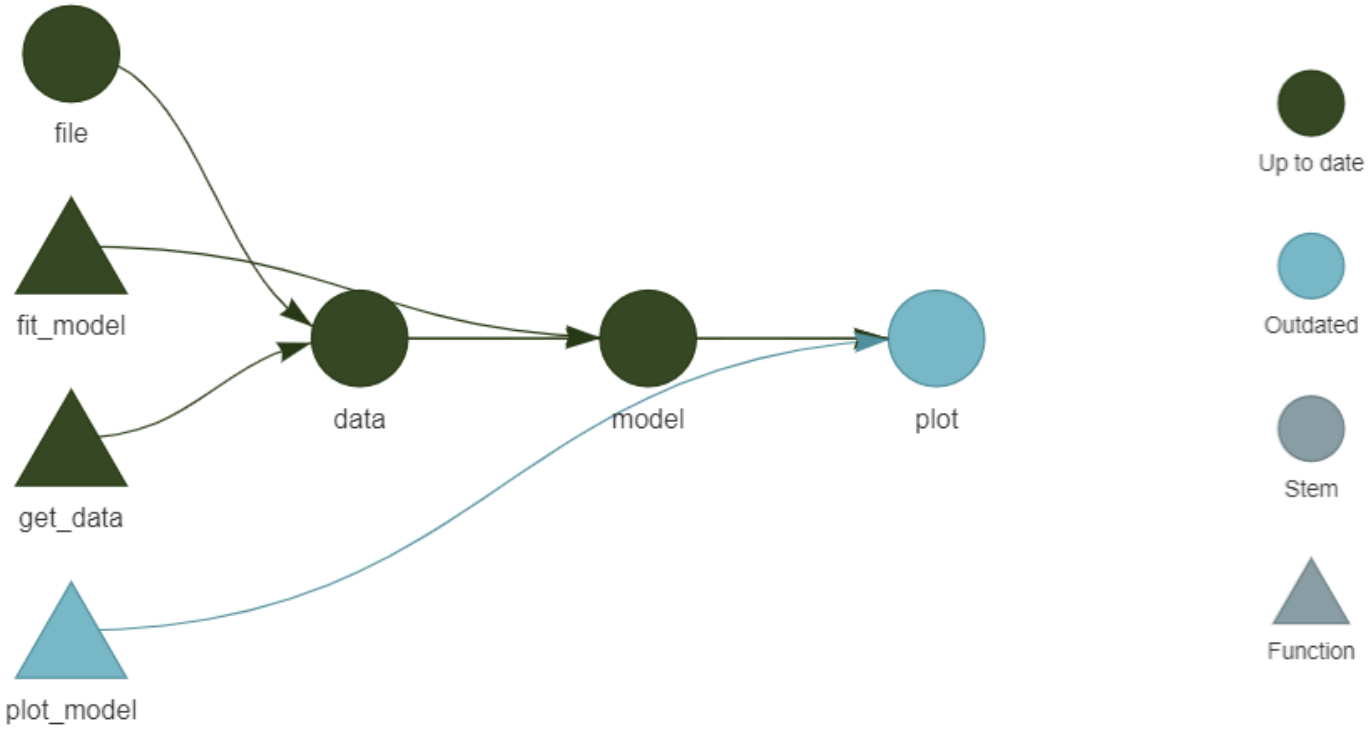
manage
data dependencies



renv
package in R

manage
package dependencies

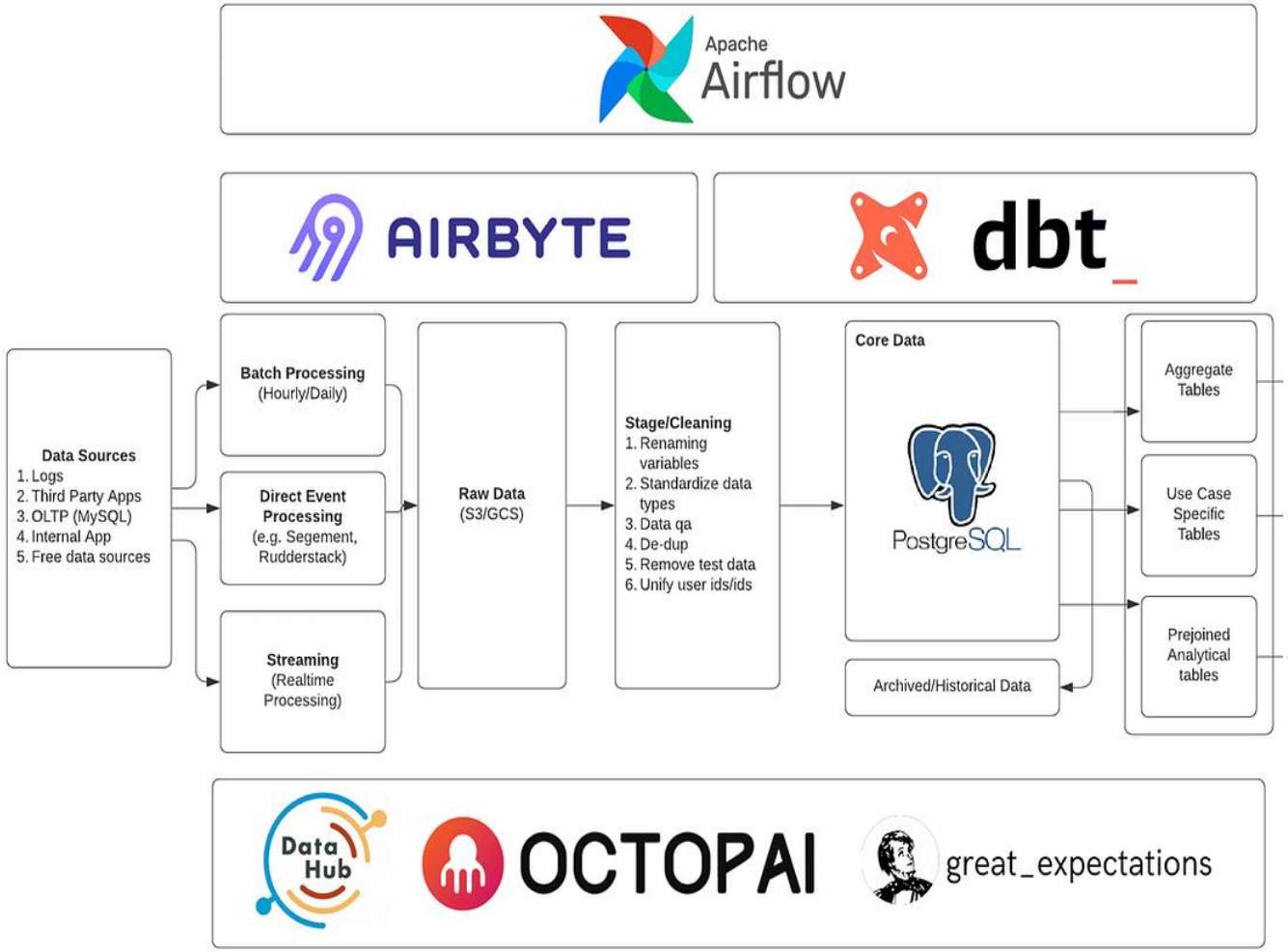
```
tar_visnetwork()
```



targets
package in R

manage
model
dependencies

Example Modern Data Engineering Stack



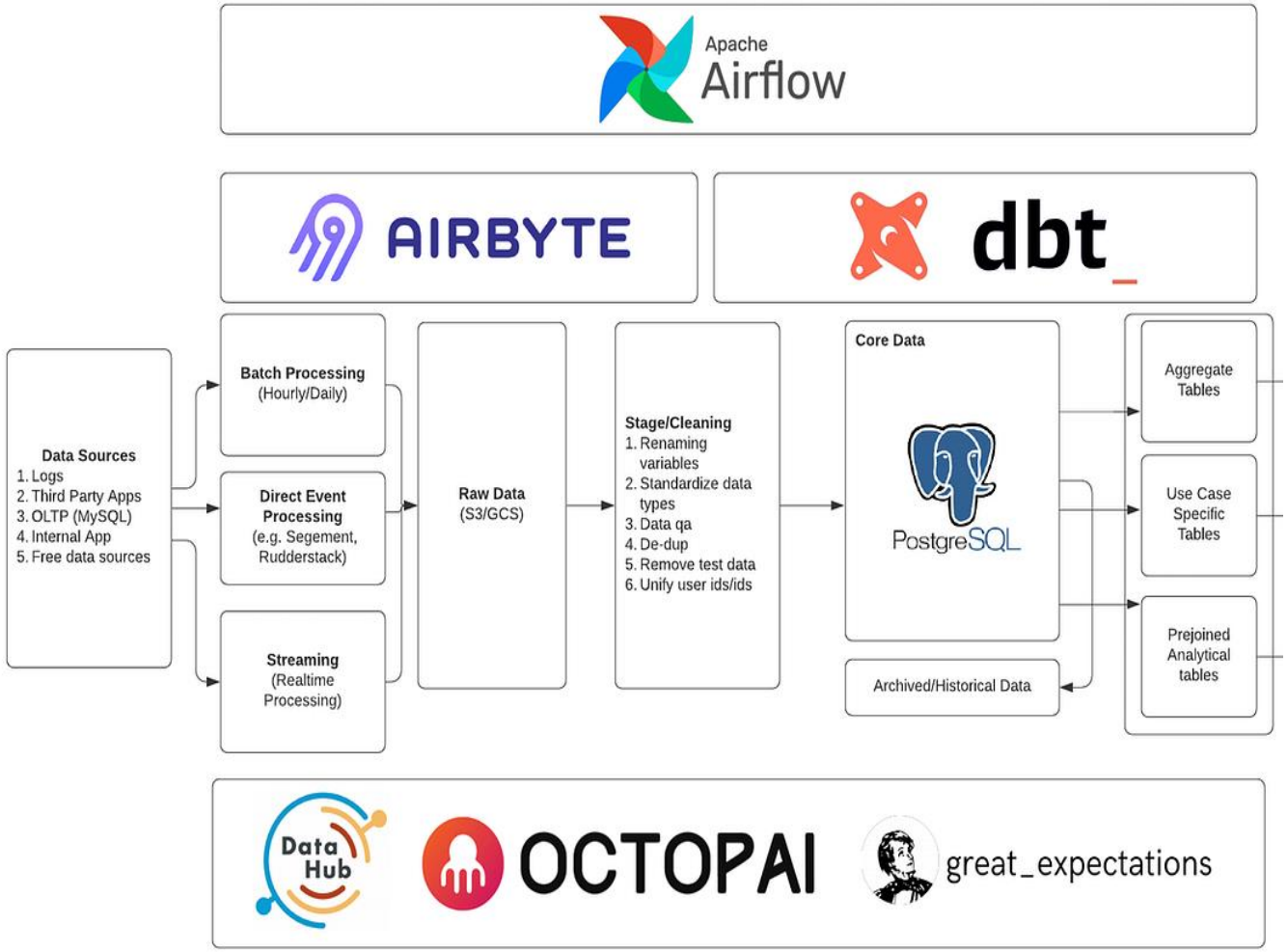
environment dependencies

myriad language & platform requirements

datagood(r)

R package to
simplify quality assurance
& automate documentation
in the data pipeline

Example Modern Data Engineering Stack



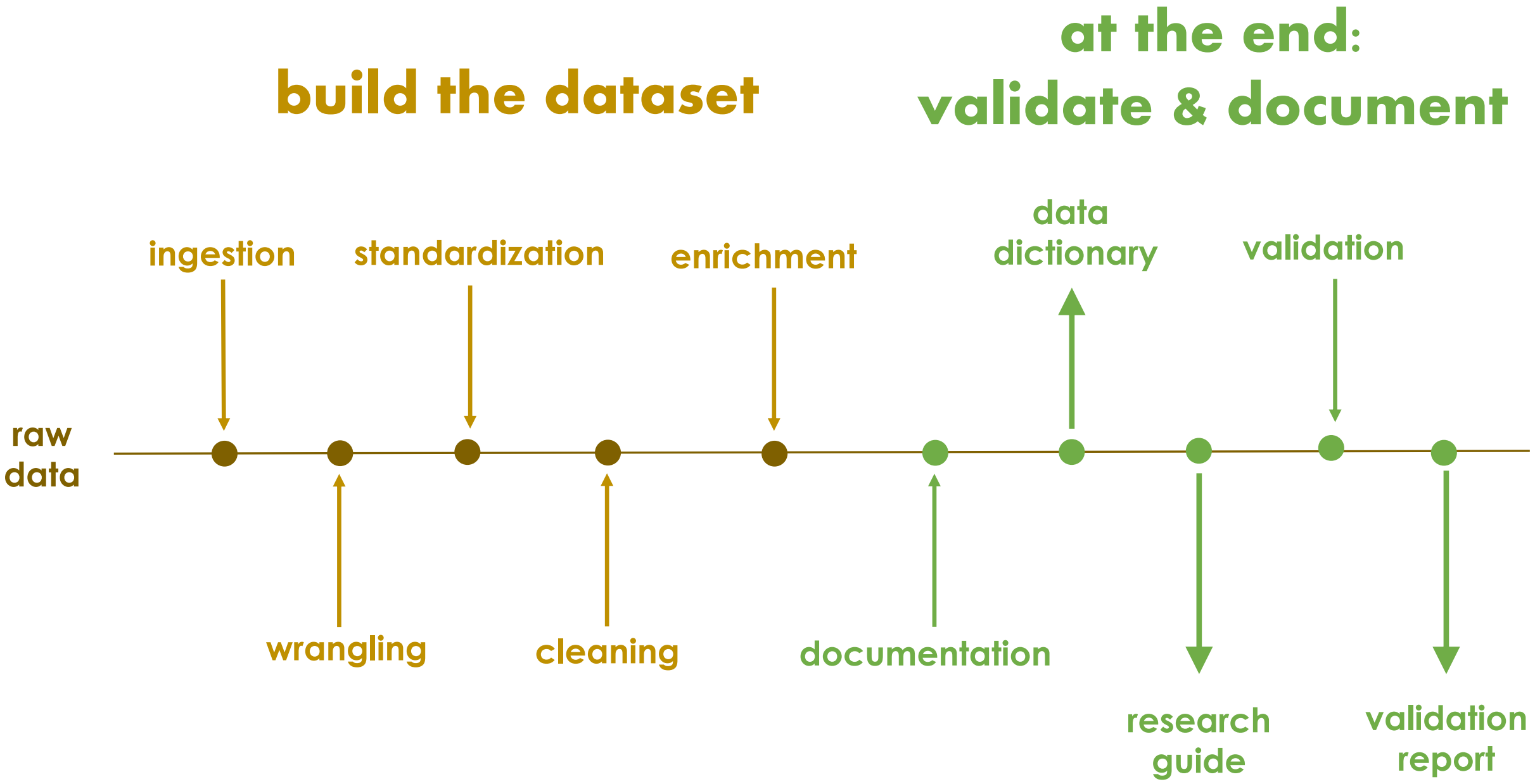
datagood(r) stack

CSV

R

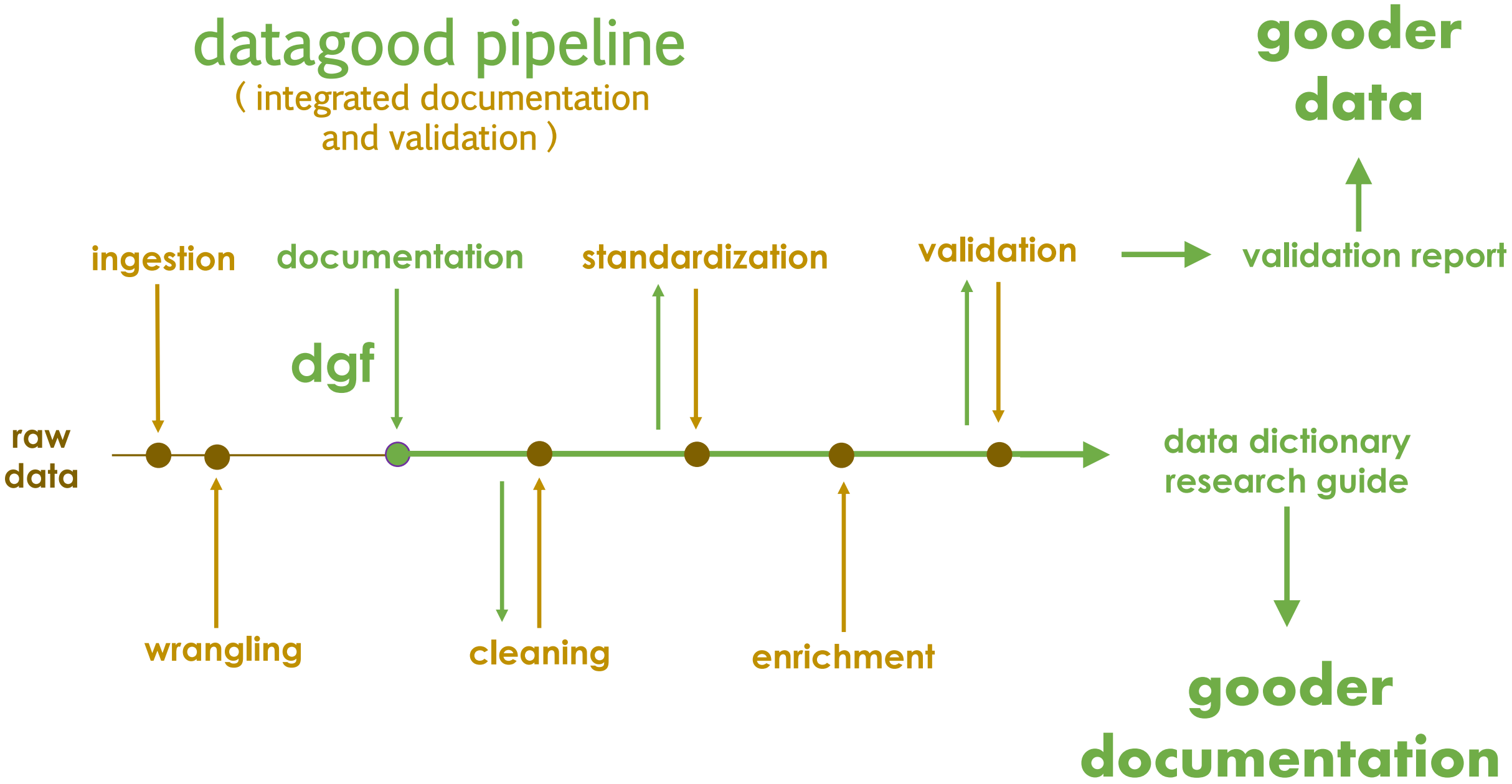
RMarkdown

TYPICAL DATA WORKFLOW


build the dataset
at the end:
validate & document
ingestion
standardization
enrichment
data
dictionary
validation
raw
data
wrangling
cleaning
documentation
research
guide
validation
report

datagood pipeline

(integrated documentation and validation)



VarName_01_X

LABEL: Ipsum is simply dummy text of the printing and typesetting industry.

DATA TYPE: numeric

SCOPE: PZ

DESCRIPTION: Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

LEVELS

FLEVEL	LABEL
AL	Alabama
AK	Alaska
AZ	Arizona
AR	Arkansas
CA	California
CO	Colorado

LOCATION CODE: SCHED-A-PART-01-LINE-01

Properties

Distinct (n)	1000
Distinct (%)	1
Missing (n)	0
Missing (%)	0

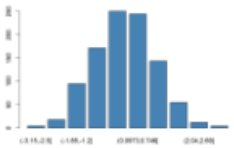
Quantiles

Q-05	-1.5918781
Q-25	-0.7427060
Q-50	-0.0388233
Q-75	0.6755618
Q-95	1.7162577

Statistics

Min	-3.3136083
Median	-0.0388233
Mean	-0.0262461
Max	3.3802229
Skew	0.1417956
Kurt	-0.0856412

Histogram



Example values

-5000	25967	111407	78480	139944
131477	17464	129319	44112	278383
46928	243061	-88203	26120	56314
136676	105615	64531	156974	-210333
34059	-12598	25034	66552	162975

data dictionary

data profiling

vname	vlabel	vdesc	vname_alias	raw_first5	raw_type	raw_convert	vtype	vtype_class	vformat_out	first5	values	f_levels	f_order	standardize	validate
EIN	EIN	EIN	EIN	10018323 ;; 10018327 ;; 10018330 ;; 10019705 ;; 100215	numeric	as.character()	character	ein		10018323 ;; 10018327 ;; 1001	[{ "stat" : "n_missing", "EIN			as_ein()	
NCCS_ACCPER	Accounting Period	Tax period end date	ACCPER	5 ;; 12 ;; 4 ;; 3 ;; 6	factor	as_mm()	date		MM	5 ;; 12 ;; 4 ;; 3 ;; 6	[{ "stat" : "n_missing", "AC	[{ "f_level" : "1" , "label"			col_vals_in_set(columns =
BMF_ACTIV1	Activity Code	IRS Activity Code 1	ACTIV1	307 ;; 260 ;; 205 ;; 280 ;; 200	numeric	as.factor()	factor			307 ;; 260 ;; 205 ;; 280 ;; 200	[{ "stat" : "n_missing", "AC				
BMF_ACTIV2	Activity Code	IRS Activity Code 2	ACTIV2	308 ;; 279 ;; 0 ;; 402 ;; 265	numeric	as.factor()	factor			308 ;; 279 ;; 0 ;; 402 ;; 265	[{ "stat" : "n_missing", "AC				
BMF_ACTIV3	Activity Code	IRS Activity Code 3	ACTIV3	0 ;; 403 ;; 319 ;; 279 ;; 317	numeric	as.factor()	factor			0 ;; 403 ;; 319 ;; 279 ;; 317	[{ "stat" : "n_missing", "AC				
F9_00_ORG_ADDR_L1	Address	Organization street address line 1	ADDRESS	PO BOX 801 ;; 120 DRUMMOND AVEN... ;; 5 VERTI DF	character		character	address.line1		PO BOX 801 ;; 120 DRUMMC	[{ "stat" : "n_missing", "AD			toupper(x)	
BMF_AFCD	Group code	Group exemption number	AFCD	3 ;; 6 ;; 3 ;; 0 ;; 2	factor		factor			3 ;; 6 ;; 3 ;; 0 ;; 2					col_vals_in_set(c
F9_10_ASSET_TOT_BOY	ASS_BOY	Total assets - beginning of year	ASS_BOY	402383 ;; 1026253 ;; 1878445 ;; 651884 ;; 1143050	numeric		numeric			402383 ;; 1026253 ;; 1878445	[{ "stat" : "n_missing", "AS				
F9_10_ASSET_TOT_EOY	ASS_EOY	Total assets - end of year	ASS_EOY	175678 ;; 381124 ;; 1135218 ;; 1941275 ;; 732881	numeric		numeric			175678 ;; 381124 ;; 1135218 ;; 1	[{ "stat" : "n_missing", "AS				
F9_10_LIAB_TAX_EXEMPT_BOND_BOY	BOND_BOY	Tax-exempt bond liabilities - beginning	BOND_BOY	0 ;; -500 ;; 11922 ;; 2865650 ;; 235000	numeric		numeric			0 ;; -500 ;; 11922 ;; 2865650 ;;	[{ "stat" : "n_missing", "BC				
F9_10_LIAB_TAX_EXEMPT_BOND_EOY	BOND_EOY	Tax-exempt bond liabilities - end of year	BOND_EOY	0 ;; 2430000 ;; 270000 ;; 7309 ;; 35934580	factor	as.numeric()	numeric			0 ;; 2430000 ;; 270000 ;; 730	[{ "stat" : "n_missing", "BC	[{ "f_level" : "-5268" , "l	-5268 ;; 0 ;; 361 ;; 3713 ;; 375		
CEO_CENSUSTRACT	CENSUSTRACT	Census tract	CENSUSTRACT	23031030202 ;; 23011024102 ;; 23011024200 ;; 23019	numeric		numeric			23031030202 ;; 2301102410	[{ "stat" : "n_missing", "CE				
F9_00_ORG_ADDR_CITY	CITY	Organization city	CITY	SANFORD ;; WATERVILLE ;; WINSLOW ;; HOLDEN ;; A	character		character	address.city		SANFORD ;; WATERVILLE ;;	[{ "stat" : "n_missing", "CI			toupper(x)	
BMF_CLASSCD	CLASSCD	IRS Classification code	CLASSCD	10 ;; 30 ;; 20 ;; 32 ;; 40	factor		factor			10 ;; 30 ;; 20 ;; 32 ;; 40	[{ "stat" : "n_missing", "CL	[{ "f_level" : "0" , "label"	0 ;; 1 ;; 2 ;; 3 ;; 4 ;; 10 ;; 12 ;; 13		
F9_09_EXP_COMP_DTK_TOT	COMPENS	Compensation of current officers, direct	COMPENS	15603 ;; 0 ;; 110260 ;; 47420 ;; 197859	numeric		numeric			15603 ;; 0 ;; 110260 ;; 47420 ;;	[{ "stat" : "n_missing", "CC				
F9_01_EXP_SAL_ETC_PY	COMPENSP	Salaries, other comp., employee benefit	COMPENSP	0 ;; 107600 ;; 47481 ;; 197430 ;; 72315	numeric		numeric			0 ;; 107600 ;; 47481 ;; 197430	[{ "stat" : "n_missing", "CC				
F9_08_REV_CONTR_TOT	CONT	Total contributions, gifts, grants, and oth	CONT	12048 ;; 10723 ;; 635193 ;; 2400 ;; 330639	numeric		numeric			12048 ;; 10723 ;; 635193 ;; 24	[{ "stat" : "n_missing", "CC				
CONTACT	CONTACT	Contact person (from IRS files)	CONTACT	TOM ADKINS TREAS ;; SCOTT HALLOWELL C... ;; WOI	character		character			TOM ADKINS TREAS ;; SCO	[{ "stat" : "n_missing", "CC				
F9_01_REV_CONTR_TOT_PY	CONTP	Contributions and grants - prior year	CONTP	14360 ;; 514432 ;; 2000 ;; 1007625 ;; 704306	numeric		numeric			14360 ;; 514432 ;; 2000 ;; 100	[{ "stat" : "n_missing", "CC				
DEDUCTCD	DEDUCTCD	IRS Deductibility code	DEDUCTCD	1 ;; 2 ;; 0 ;; 4	factor		factor			1 ;; 2 ;; 0 ;; 4	[{ "stat" : "n_missing", "				
F9_08		vents	DIREXP	741314 ;; 33826 ;; 0 ;; 34759 ;; 20814	numeric		numeric			741314 ;; 33826 ;; 0 ;; 34759 ;	[{ "st				
			EOSTATUS	1 ;; 12	numeric	as.factor()	factor			1 ;; 12	[{				
				0	numeric	as.factor()	factor			0					

THE DATA GOVERNANCE FILE (DGF):

A RULE-BASED APPROACH TO MANAGING DATA COMPLEXITY

Data Governance File (DGF)

Data Dictionary	Ingestion Rules	Documentation	Standardization Rules	Validation Rules

A	B	C	D
vname	**vlabel**	**vdesc**	**vname_alias**
EIN	EIN	EIN	EIN
NCCS_ACCPER	Accounting Period	Tax period end date	ACCPER
BMF_ACTIV1	Activity Code	IRS Activity Code 1	ACTIV1
BMF_ACTIV2	Activity Code	IRS Activity Code 2	ACTIV2
BMF_ACTIV3	Activity Code	IRS Activity Code 3	ACTIV3
F9_00_ORG_ADDR_L1	Address	Organization street address line 1	ADDRESS

Data Governance File (DGF)

Data Dictionary	Ingestion Rules	Documentation	Standardization Rules	Validation Rules

A	E	F	G	H
vname	**raw_first5**	**raw_type**	**raw_convert**	**vtype**
EIN	10018923 ;; 10018927 ;; 10018930 ;; 10019705 ;; 10021545	numeric	as.character()	character
NCCS_ACCPER	5 ;; 12 ;; 4 ;; 3 ;; 6	factor	as_mm()	date
BMF_ACTIV1	907 ;; 260 ;; 205 ;; 280 ;; 200	numeric	as.factor()	factor
BMF_ACTIV2	908 ;; 279 ;; 0 ;; 402 ;; 265	numeric	as.factor()	factor
BMF_ACTIV3	0 ;; 403 ;; 319 ;; 279 ;; 317	numeric	as.factor()	factor
F9_00_ORG_ADDR_L1	PO BOX 801 ;; 120 DRUMMOND AVEN... ;; 5 VERTI DR ;; PO BC	character		character

data types + formats

```
 [1] "BooleanType"
 [3] "CheckboxType"
 [4] "USAmountNNType"
 [5] "StringType"
 [6] "CountryType"
 [7] "StateType"
 [8] "BusinessNameControlType"
 [9] "TextType ;; CityType"
[10] "InCareOfNameType"
[11] "StreetAddressType"
[12] "TextType ;; StateType"
[13] "TextType ;; ZIPCodeType"
[14] "EINType"
[15] "BusinessNameLine1Type"
[16] "BusinessNameLine2Type"
[17] "PhoneNumberType"
[18] "LineExplanationType"
[19] "PersonNameType"
[20] "TimestampType"
[21] "TextType"
[22] "DateType"
[23] "YearType"
[24] "ShortExplanationType"
[25] "IntegerNNType"
[26] "USAmountType"
[27] "USAmountNNType ;; USAmountType"
[28] "PTINType"
[29] "PersonTitleType"
[30] "ExplanationType"
[31] "CheckboxType ;; BooleanType"
[32] "CountType"
[33] "TextType ;; StateType ;; CountryType"
[34] "xsd:decimal ;; LargeRatioType ;; IntegerNNType"
[35] "xsd:decimal"
[36] "CountryType ;; CityType ;; TextType"
[37] "CountryType ;; StreetAddressType"
[38] "CountryType ;; StateType ;; TextType"
[39] "CountryType ;; ZIPCodeType ;; TextType"
[40] "CountryType ;; BusinessNameLine2Type"
[41] "IntegerNNType ;; CountType"
[42] "USAmountType ;; USAmountNNType"
[43] "CityType ;; TextType"
[44] "StateType ;; TextType"
[45] "ZIPCodeType ;; TextType"
[46] "LargeRatioType"
[47] "CityType"
[48] "StringType ;; ShortDescriptionType"
[49] "RatioType"
[50] "DecimalNNType"
[51] "StateType ;; StringType"
[52] "ZIPCodeType"
[53] "AlphaNumericType"
[54] "Count2Type"
[55] "ShortDescriptionType"
[56] "CUSIPNumberType"
[57] "IntegerNNType ;; LargeRatioType"
[58] "TextType ;; StringType"
```

dgf_raw + ingest(): date example

(R easily recognizes this format as a Date, not character, when loading a CSV)

"12/27/81"	1981-12-27	"Dec 27, 1981"
"9/8/54"	1954-09-08	"Sep 08, 1954"
"8/17/72"	1972-08-17	"Aug 17, 1972"

dgf_raw_type
character

dgf_raw_convert
as.date(mm/dd/yy)

dgf_type
date

dgf_type_class
(blank)

dgf_format_out
mmm dd, yyyy

df.v2

```
update_dgf( df.v2 )
```

```
format( df, dgf,
 out="public.csv" )
```

```
ingest_raw( df.raw, dgf )
```

After ingest() the data is stored in the format that is easiest to use.

The **dgf_raw_type** and the conversion rule (**dgf_raw_convert**) defines how to handle the raw version of the data.

phone number

1234567890	→	123-456-7890
9087654321		908-765-4321
8002437866		800-243-7866

dgf_type	dgf_type_class	dgf_format_out
numeric	phone	phone

```
format( df, dgf,
  out="dataf.csv" )
```

The **`dgf_format_out`** value allows you reformat data in a way that is better for public consumption but a pain for analysts.

currency example

Format preferred for analysis.

raw data	df.v2 … df.vX	df.public
$ 1,225.33	1225.326	$1,225.33
$ 67,894.00	67894	$67,894.00
$ 23.23	23.234643	$23.23

dgf_raw_type	dgf_raw_convert	dgf_type	dgf_type_class	dgf_format_out
character	as.number	numeric	currency	usd

from.usd?

Can use existing R expressions or write your own. For example, `as.number()` would regex all non-numeric values then use `as.numeric()`, thus removing the dollar sign and comma.

We want to preserve `raw_convert` rules because we will likely use data from the same source again or we might be cleaning a new wave of the same data. After ingestion, though, we store df.v2 and subsequent iterations as the desired data type.

data type & type class

We use **data type** to describe the storage type in R (primitives). Database schemas have a broader view, which is called **dgf_type_class** in the datagood framework.

Examples of data 'types' used by IRS 990 efiling database schemas. These can be assigned as class attributes in the DGF if they make the data easier to work with (enables future import functionality for building data dictionaries from schemas).

Ideally there would be a library of import and standardization functions available for different data types. For example, the r-usps package can standardize addresses. A zipcode function adds leading zeros back to ensure all are 5 digits. There is an http function that normalizes web addresses. Etc.

The type_class enables functionality, but not sure how useful it would be without a library of tools for different types – mostly it gives the user options if they have complex data that requires nuance.

Data Governance File (DGF)

Data Dictionary	Ingestion Rules	**Documentation**	Standardization Rules	Validation Rules

M	N
f_levels	**f_order**
[{ "f_level" : 1 ;; 2 ;; 3 ;; 4 ;	

```
[
  {  "f_level" :  "1"  ,   "label" :  "January"    },
  {  "f_level" :  "2"  ,   "label" :  "February"   },
  {  "f_level" :  "3"  ,   "label" :  "March"      },
  {  "f_level" :  "4"  ,   "label" :  "April"      },
  {  "f_level" :  "5"  ,   "label" :  "May"        },
  {  "f_level" :  "6"  ,   "label" :  "June"       },
  {  "f_level" :  "7"  ,   "label" :  "July"       },
  {  "f_level" :  "8"  ,   "label" :  "August"     },
  {  "f_level" :  "9"  ,   "label" :  "September"  },
  {  "f_level" :  "10"  ,  "label" :  "October"    },
  {  "f_level" :  "11"  ,  "label" :  "November"   },
  {  "f_level" :  "12"  ,  "label" :  "December"   }
]
```

VarName_02_L2

Ipsum is simply dummy text of the printing and typesetting industry.

DATA TYPE:	**factor**
SCOPE:	**PC**
LENGTH:	**24**

DEFINITION: Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

LEVELS:

A	goat
B	monkey
C	turtle
D	horse

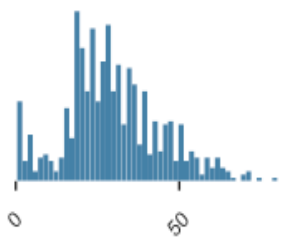
data dictionary format

Data Governance File (DGF)

Data Dictionary	Ingestion Rules	**Documentation**	Standardization Rules	Validation Rules

K	L
first5	**values**
10018923 ;; 10018927 ;; 100189	[{ "stat" : "n_missing", "EIN" : "C
5 ;; 12 ;; 4 ;; 3 ;; 6	[{ "stat" : "n_missing", "ACCPER
907 ;; 260 ;; 205 ;; 280 ;; 200	[{ "stat" : "n_missing", "ACTIV1'
908 ;; 279 ;; 0 ;; 402 ;; 265	[{ "stat" : "n_missing", "ACTIV2'
0 ;; 403 ;; 319 ;; 279 ;; 317	[{ "stat" : "n_missing", "ACTIV3'
PO BOX 801 ;; 120 DRUMMOND	[{ "stat" : "n_missing", "ADDRES

Distinct count	89	Mean	29.69911765
Unique (%)	10.0%	Minimum	0.42
Missing (%)	19.9%	Maximum	80
Missing (n)	177	Zeros (%)	0.0%
Infinite (%)	0.0%		
Infinite (n)	0		

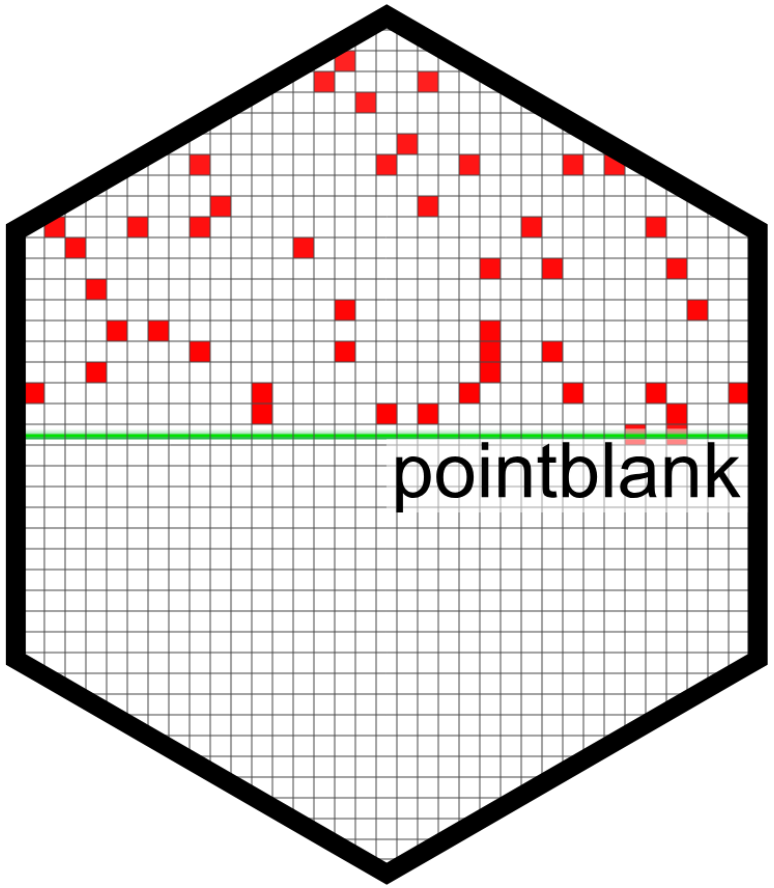


Toggle details

Data Governance File (DGF)

Data Dictionary	Ingestion Rules	Documentation	Standardization Rules	Validation Rules

A	O	P
vname	**standardize**	**validate**
EIN	as_ein()	
NCCS_ACCPER		col_vals_in_set(columns = vars(x), set = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12))
BMF_ACTIV1		
BMF_ACTIV2		
BMF_ACTIV3		
F9_00_ORG_ADDR_L1	toupper(x)	
BMF_AFCD		col_vals_in_set(columns = vars(x), set = c(0, 1, 2, 3, 6, 7, 8, 9))



https://rich-iannone.github.io/pointblank/reference/index.html

pointblank 0.11.0.9000 Articles Reference News

Are column data not equal to a fixed value or data in another column?

`col_vals_gte()` `expect_col_vals_gte()` `test_col_vals_gte()`
Are column data greater than or equal to a fixed value or data in another column?

`col_vals_gt()` `expect_col_vals_gt()` `test_col_vals_gt()`
Are column data greater than a fixed value or data in another column?

`col_vals_between()` `expect_col_vals_between()` `test_col_vals_between()`
Do column data lie between two specified values or data in other columns?

`col_vals_not_between()` `expect_col_vals_not_between()` `test_col_vals_not_between()`
Do column data lie outside of two specified values or data in other columns?

`col_vals_in_set()` `expect_col_vals_in_set()` `test_col_vals_in_set()`
Are column data part of a specified set of values?

`col_vals_not_in_set()` `expect_col_vals_not_in_set()` `test_col_vals_not_in_set()`
Are data not part of a specified set of values?

`col_vals_make_set()` `expect_col_vals_make_set()` `test_col_vals_make_set()`
Is a set of values entirely accounted for in a column of values?

`col_vals_make_subset()` `expect_col_vals_make_subset()` `test_col_vals_make_subset()`
Is a set of values a subset of a column of values?

VarName_01_X

data validation report

TIBBLE	small_table	WARN	0.10	STOP	0.20	NOTIFY	–

	STEP	COLUMNS	VALUES	TBL	EVAL	···	PASS	FAIL	W	S	N	EXT
1	col_is_posix()	date_time	—	→	✓	1	1 1.00	0 0.00	○	○	—	—
2	col_vals_in_set()	f	low, mid	→	✓	13	7 0.54	6 0.46	●	●	—	CSV
3	col_vals_lt()	a	7	→	✓	13	11 0.85	2 0.15	●	○	—	CSV
4	col_vals_regex()	b	^[0-9]-[a-w]{3}…	→	✓	13	6 0.46	7 0.54	●	●	—	CSV
5	col_vals_between()	d	[0, 4,000]	→	✓	13	12 0.92	1 0.08	●	○	—	CSV

2020-11-03 14:16:33 EST	< 1 s	2020-11-03 14:16:33 EST

CUSTOMIZING LAYOUTS

VarName_01_X

Ipsum is simply dummy text of the printing and typesetting industry.

DATA TYPE: **numeric**
SCOPE: **PZ**
LENGTH: **6**

DEFINITION: Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

VarName_02_L2

div1

div2 Ipsum is simply dummy text of the printing and typesetting industry.

DATA TYPE: **factor**
SCOPE: **PC**
LENGTH: **24**

div3

DEFINITION: Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

LEVELS:

A	goat
B	monkey
C	turtle
D	horse

div4

data dictionary format

```
create_dd( dgf )
+
layout
```

VarName_01_X

div1

LABEL: Ipsum is simply dummy text of the printing and typesetting industry.

div2

DATA TYPE: numeric

SCOPE: PZ

div3

DESCRIPTION: Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

LEVELS

FLEVEL	LABEL
AL	Alabama
AK	Alaska
AZ	Arizona
AR	Arkansas
CA	California
CO	Colorado

div4

LOCATION CODE: SCHED-A-PART-01-LINE-01

Properties

Distin	
Distin	
Missing (n)	0
Missing (%)	0

div5

Quantiles

Q-05	-1.5
Q-25	-0.7
Q-50	-0.0388233
Q-75	0.6755618
Q-95	1.7162577

div6

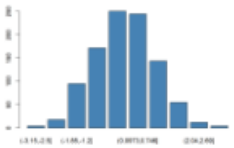
Statistics

Min	-3.31
Median	-0.038
Mean	-0.0262461
Max	3.3802229
Skew	0.1417956
Kurt	-0.0856412

div7

Histogram

div8



Example values

-5000	25967	111407	78480	139944
131477	17464	129319	44112	278383
46928	243061	-88203	26120	56314
136676	105615	64531	156974	-210333
34059	-12598	25034	66552	162975

div9

research guide format

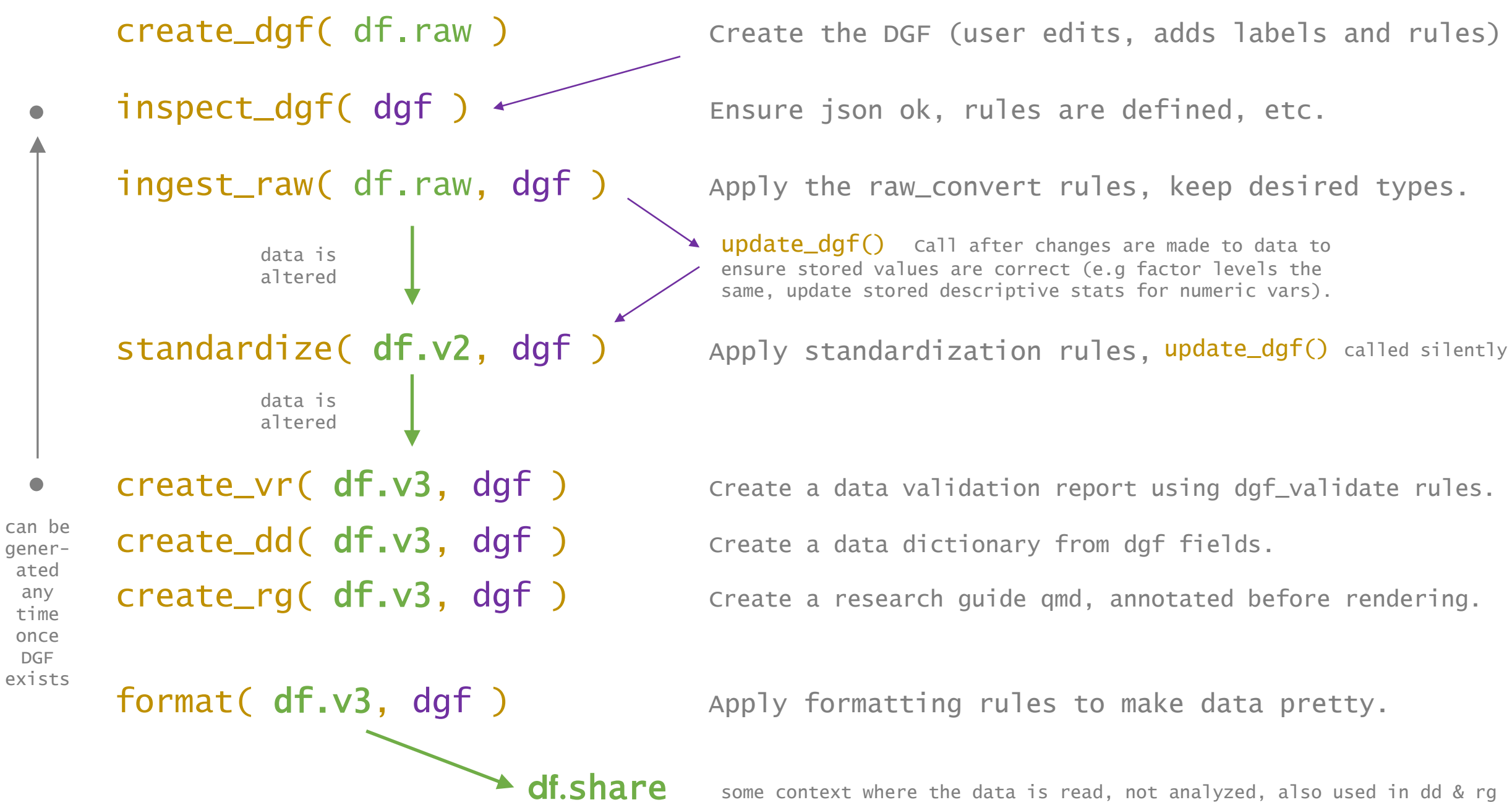
data profiling

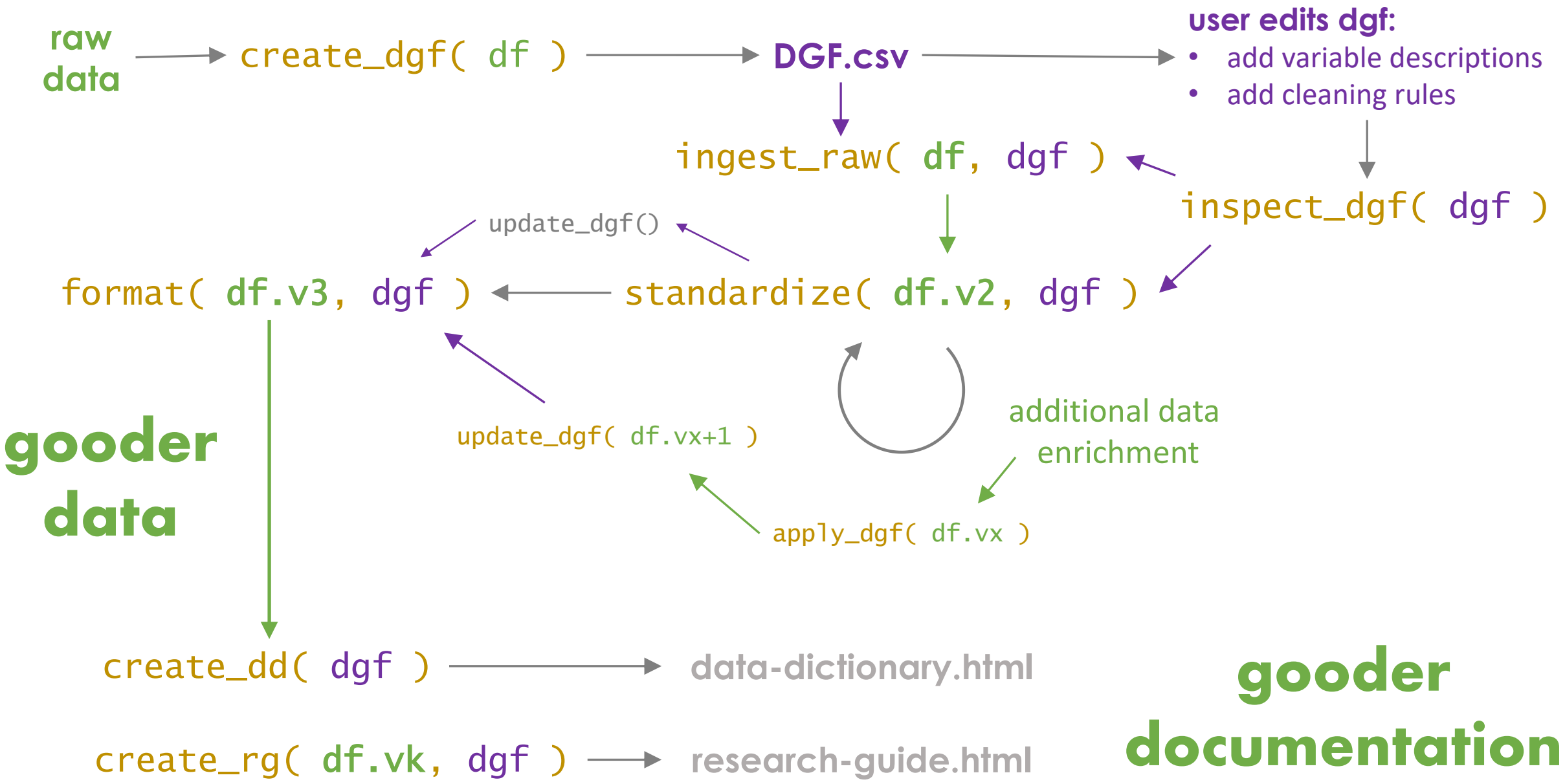
layout <-

```
c( "div2 ;; vlabel  ;; LABEL         ;; v_to_txt",
   "div3 ;; vtype   ;; DATA TYPE     ;; v_to_txt",
   "div3 ;; scope   ;; SCOPE         ;; v_to_txt",
   "div4 ;; desc    ;; DESCRIPTION   ;; v_to_txt",
   "div4 ;; flevels ;; LEVELS        ;; f_to_txt",
   "div4 ;; glevels ;; ''            ;; v_to_txt",
   "div4 ;; loc     ;; LOCATION CODE ;; v_to_txt",
   "div5 ;; v       ;; STATS         ;; get_properties"  )
```

```
|DIV  |VARIABLE  |LABEL         |FORMATTING FUNCTION  |
|:----|:---------|:-------------|:--------------------|
|div2 |vlabel    |LABEL         |v_to_txt             |
|div3 |vtype     |DATA TYPE     |v_to_txt             |
|div3 |scope     |SCOPE         |v_to_txt             |
|div4 |desc      |DESCRIPTION   |v_to_txt             |
|div4 |flevels   |LEVELS        |f_to_txt             |
|div4 |glevels   |''            |v_to_txt             |
|div4 |loc       |LOCATION CODE |v_to_txt             |
|div5 |v         |LABEL         |get_properties       |
```

WORKFLOW


create_dgf(df.raw)
Create the DGF (user edits, adds labels and rules)
inspect_dgf(dgf)
Ensure json ok, rules are defined, etc.
ingest_raw(df.raw, dgf)
Apply the raw_convert rules, keep desired types.
data is altered
update_dgf() Call after changes are made to data to ensure stored values are correct (e.g factor levels the same, update stored descriptive stats for numeric vars).
standardize(df.v2, dgf)
Apply standardization rules, update_dgf() called silently
data is altered
create_vr(df.v3, dgf)
Create a data validation report using dgf_validate rules.
create_dd(df.v3, dgf)
Create a data dictionary from dgf fields.
create_rg(df.v3, dgf)
Create a research guide qmd, annotated before rendering.
can be generated any time once DGF exists
format(df.v3, dgf)
Apply formatting rules to make data pretty.
df.share
some context where the data is read, not analyzed, also used in dd & rg


raw data
create_dgf(df)
DGF.csv
user edits dgf:
• add variable descriptions
• add cleaning rules
ingest_raw(df, dgf)
inspect_dgf(dgf)
update_dgf()
format(df.v3, dgf)
standardize(df.v2, dgf)
update_dgf(df.vx+1)
additional data enrichment
apply_dgf(df.vx)
gooder data
create_dd(dgf)
data-dictionary.html
create_rg(df.vk, dgf)
research-guide.html
gooder documentation

data dictionary format

VarName_01_X

Ipsum is simply dummy text of the printing and typesetting industry.

DATA TYPE: **numeric**
SCOPE: **PZ**
LENGTH: **6**

DEFINITION: Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

VarName_02_L2

Ipsum is simply dummy text of the printing and typesetting industry.

DATA TYPE: **factor**
SCOPE: **PC**
LENGTH: **24**

DEFINITION: Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

LEVELS:

A	goat
B	monkey
C	turtle
D	horse

thank you