R4DS

Cohort 4
Wed 6:00 – 7:00 US Central
Twitter: @Rspjut

5-MINUTE ICE BREAKER

What social media platforms do you use most?

AGENDA

- 5-Minute Ice breaker
- Quick Housekeeping Reminders
- Chapter 12 – Tidy Data
- Next Week
- Getting Help

QUICK HOUSEKEEPING REMINDERS

- Video camera is optional, but encouraged.
- I purposely err on the side of going fast. Slowing me down does not hurt my feelings.
- Take time to learn the theory (Grammar of Graphics, Tidy Data whitepaper, Relational Database theory, Appropriate Visualization Types, etc.).
- Please do the chapter exercises. Second-best learning opportunity!
- Please plan on teaching one of the lessons. Best learning opportunity!

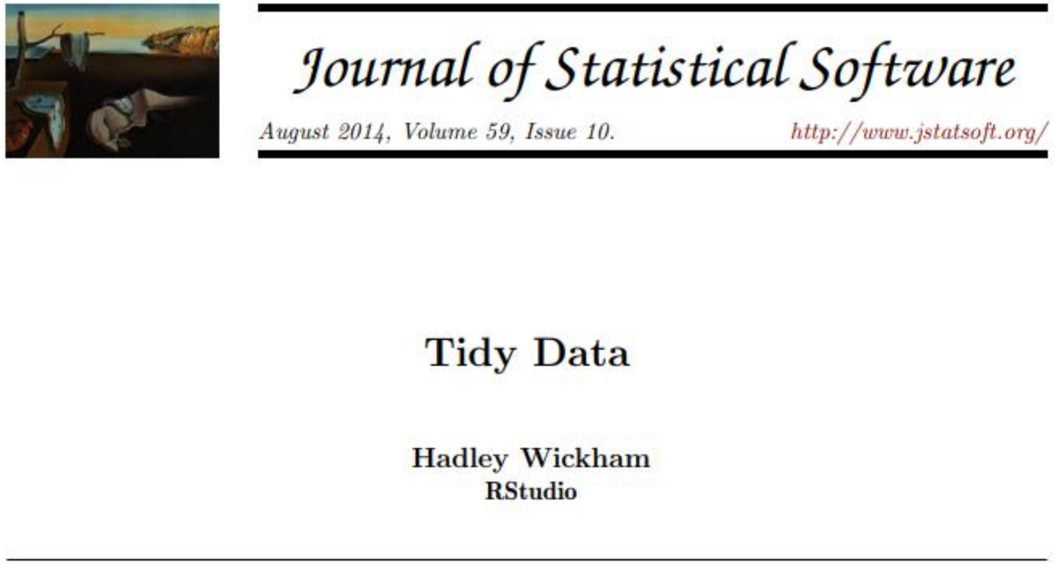
TIDY DATA: THEORY AND PRACTICE

“Happy families are all alike; every unhappy family is unhappy in its own way.” — Leo Tolstoy

“Tidy datasets are all alike, but every messy dataset is messy in its own way.” — Hadley Wickham

Hadley

Not Hadley



Journal of Statistical Software

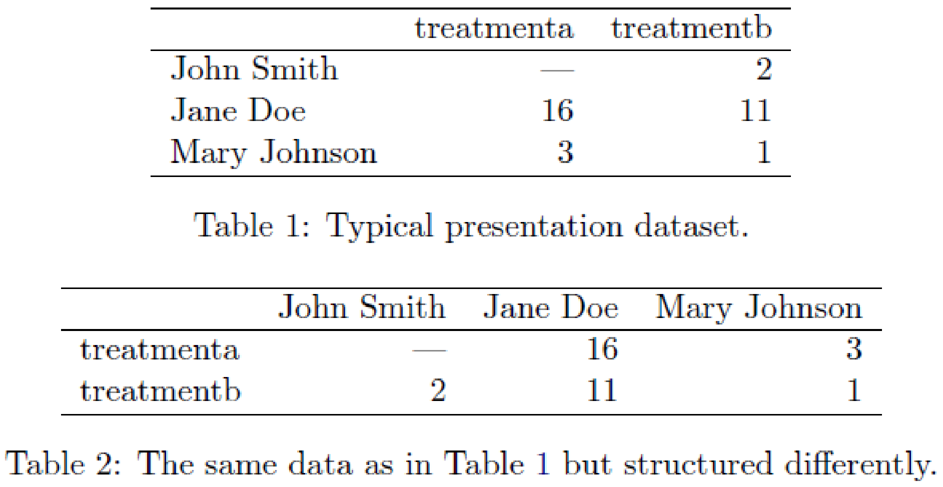
August 2014, Volume 59, Issue 10. http://www.jstatsoft.org/

Tidy Data

Hadley Wickham
RStudio

A huge amount of effort is spent cleaning data to get it ready for analysis, but there has been little research on how to make data cleaning as easy and effective as possible. This paper tackles a small, but important, component of data cleaning: data tidying. Tidy datasets are easy to manipulate, model and visualize, and have a specific structure: each variable is a column, each observation is a row, and each type of observational unit is a table. This framework makes it easy to tidy messy datasets because only a small set of tools are needed to deal with a wide range of un-tidy datasets. This structure also makes it easier to develop tidy tools for data analysis, tools that both input and output tidy datasets. The advantages of a consistent data structure and matching tools are demonstrated with a case study free from mundane data manipulation chores.

TIDY DATA: THEORY AND PRACTICE

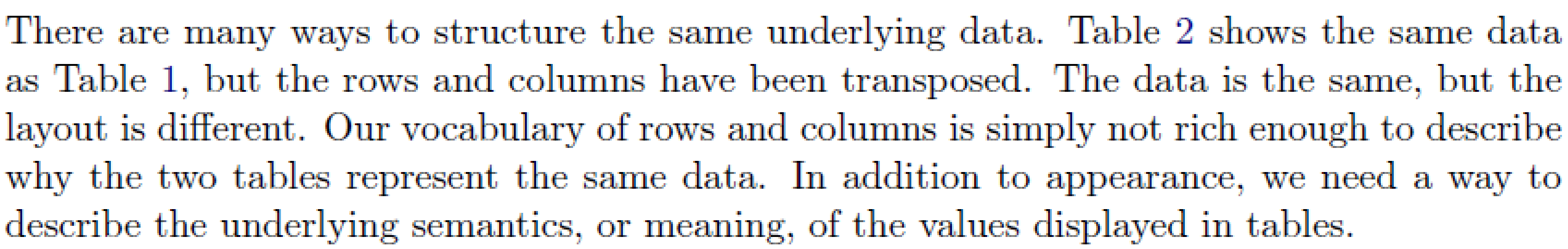


	treatmenta	treatmentb
John Smith	—	2
Jane Doe	16	11
Mary Johnson	3	1

Table 1: Typical presentation dataset.

	John Smith	Jane Doe	Mary Johnson
treatmenta	—	16	3
treatmentb	2	11	1

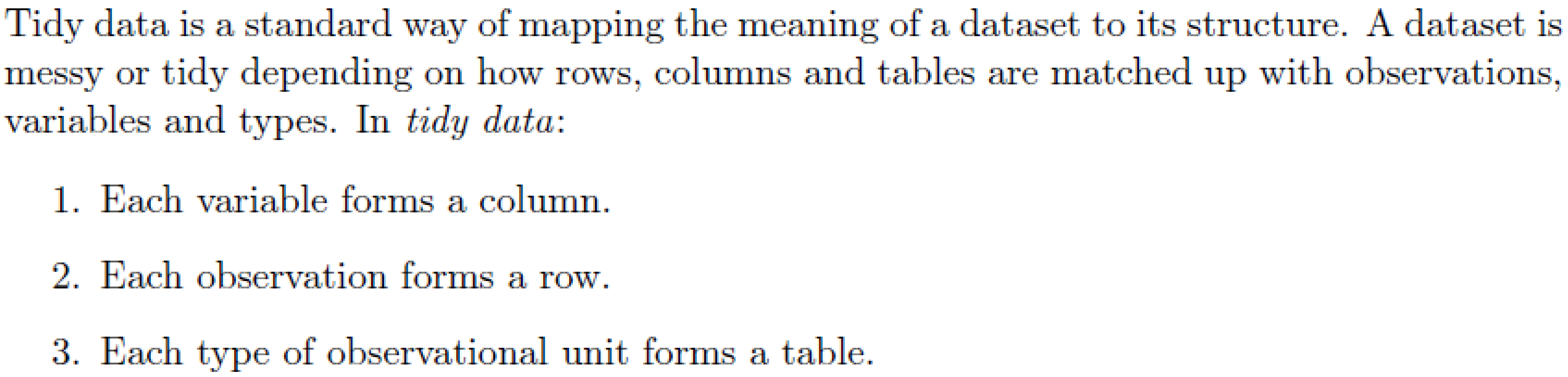
Table 2: The same data as in Table 1 but structured differently.



There are many ways to structure the same underlying data. Table 2 shows the same data as Table 1, but the rows and columns have been transposed. The data is the same, but the layout is different. Our vocabulary of rows and columns is simply not rich enough to describe why the two tables represent the same data. In addition to appearance, we need a way to describe the underlying semantics, or meaning, of the values displayed in tables.

Saying “row” or “column” presupposes a data structure, and that this data structure is known to the audience. By definition, a messy data set doesn’t have this structure, and it’s not safe to assume the structure is known.

TIDY DATA: THEORY AND PRACTICE

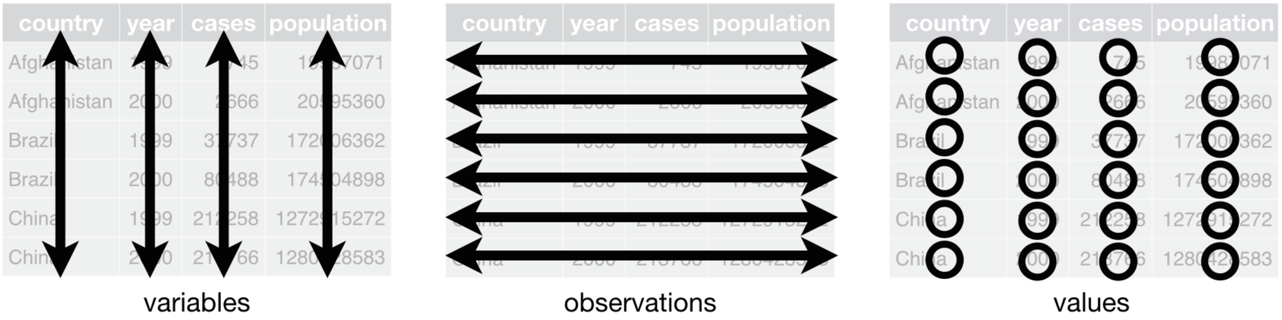


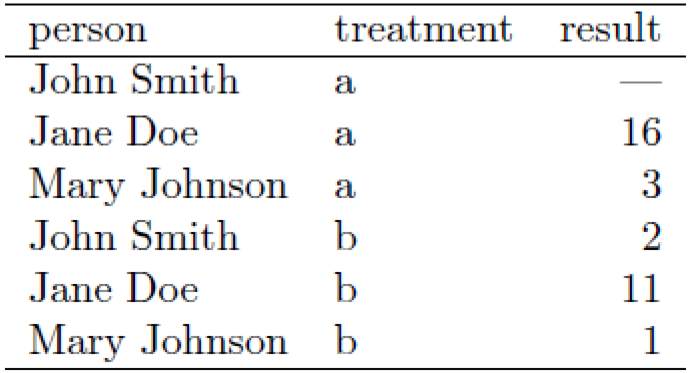
Tidy data is a standard way of mapping the meaning of a dataset to its structure. A dataset is messy or tidy depending on how rows, columns and tables are matched up with observations, variables and types. In *tidy data*:

1. Each variable forms a column.

2. Each observation forms a row.

3. Each type of observational unit forms a table.



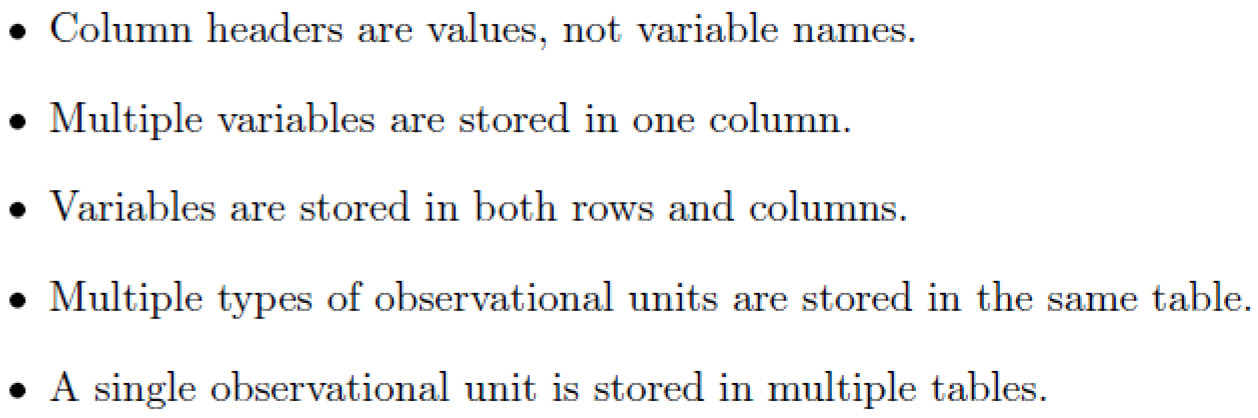


person	treatment	result
John Smith	a	—
Jane Doe	a	16
Mary Johnson	a	3
John Smith	b	2
Jane Doe	b	11
Mary Johnson	b	1

> *This layout ensures that values of different variables from the same observation are always paired.*
> *– Hadley Wickham*

TIDY DATA: THEORY AND PRACTICE

Five Common Problems With Messy Datasets

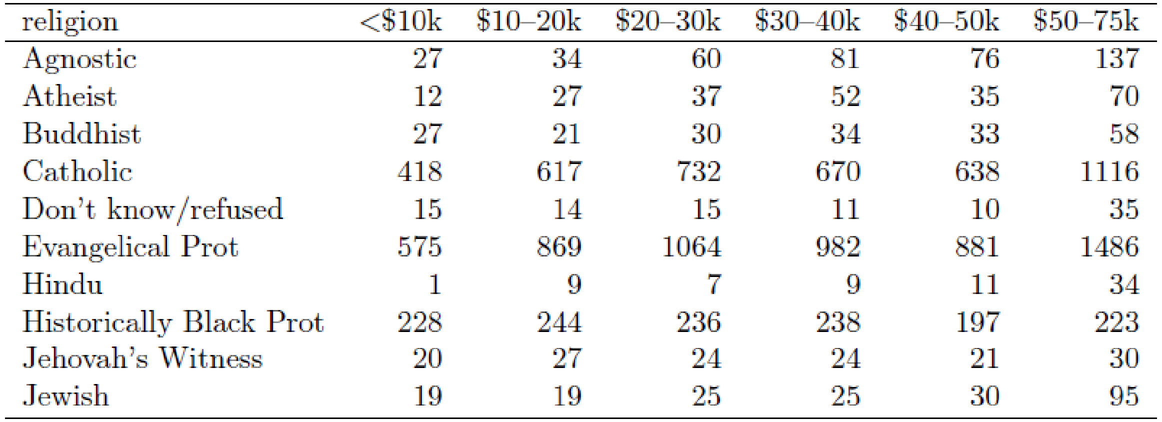


- Column headers are values, not variable names.
- Multiple variables are stored in one column.
- Variables are stored in both rows and columns.
- Multiple types of observational units are stored in the same table.
- A single observational unit is stored in multiple tables.

TIDY DATA: THEORY AND PRACTICE

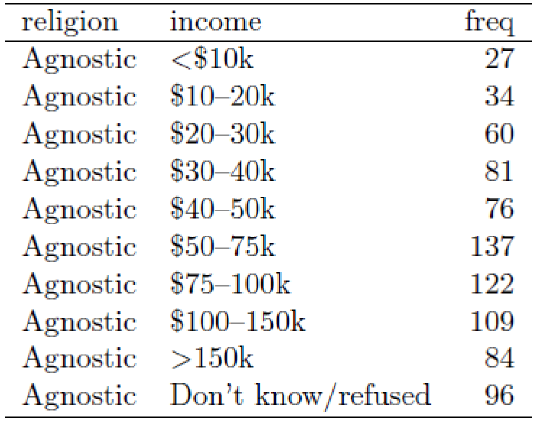
COMMON PROBLEM #1: Column Headers are Values, not Variable Names

Messy Version



religion	<$10k	$10–20k	$20–30k	$30–40k	$40–50k	$50–75k
Agnostic	27	34	60	81	76	137
Atheist	12	27	37	52	35	70
Buddhist	27	21	30	34	33	58
Catholic	418	617	732	670	638	1116
Don't know/refused	15	14	15	11	10	35
Evangelical Prot	575	869	1064	982	881	1486
Hindu	1	9	7	9	11	34
Historically Black Prot	228	244	236	238	197	223
Jehovah's Witness	20	27	24	24	21	30
Jewish	19	19	25	25	30	95

Tidy Version

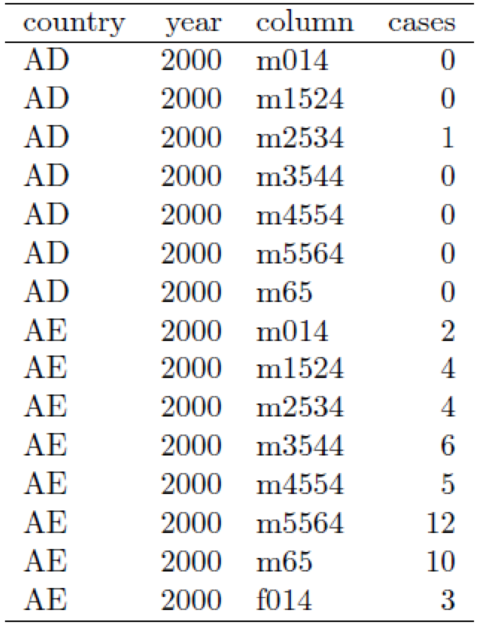


religion	income	freq
Agnostic	<$10k	27
Agnostic	$10–20k	34
Agnostic	$20–30k	60
Agnostic	$30–40k	81
Agnostic	$40–50k	76
Agnostic	$50–75k	137
Agnostic	$75–100k	122
Agnostic	$100–150k	109
Agnostic	>150k	84
Agnostic	Don't know/refused	96

TIDY DATA: THEORY AND PRACTICE

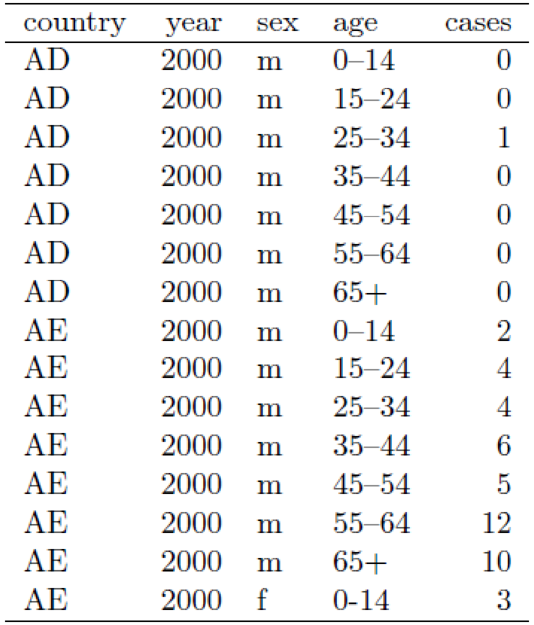
COMMON PROBLEM #2: Multiple Variables are Stored (or Encoded) in One Column

Messy Version



country	year	column	cases
AD	2000	m014	0
AD	2000	m1524	0
AD	2000	m2534	1
AD	2000	m3544	0
AD	2000	m4554	0
AD	2000	m5564	0
AD	2000	m65	0
AE	2000	m014	2
AE	2000	m1524	4
AE	2000	m2534	4
AE	2000	m3544	6
AE	2000	m4554	5
AE	2000	m5564	12
AE	2000	m65	10
AE	2000	f014	3

Tidy Version



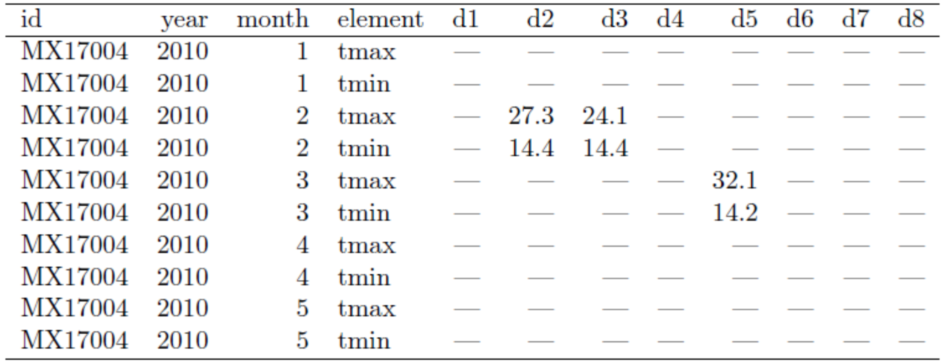
country	year	sex	age	cases
AD	2000	m	0–14	0
AD	2000	m	15–24	0
AD	2000	m	25–34	1
AD	2000	m	35–44	0
AD	2000	m	45–54	0
AD	2000	m	55–64	0
AD	2000	m	65+	0
AE	2000	m	0–14	2
AE	2000	m	15–24	4
AE	2000	m	25–34	4
AE	2000	m	35–44	6
AE	2000	m	45–54	5
AE	2000	m	55–64	12
AE	2000	m	65+	10
AE	2000	f	0-14	3

TIDY DATA: THEORY AND PRACTICE

COMMON PROBLEM #3: Variables are Stored in Both Rows and Columns

Messy Version

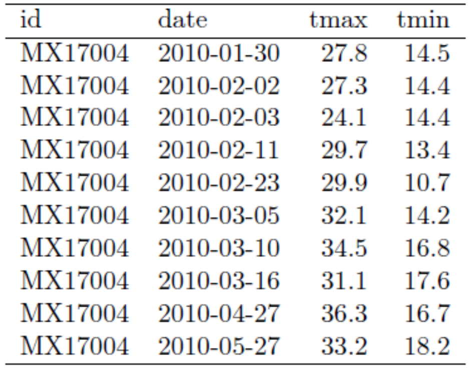
Variable



id	year	month	element	d1	d2	d3	d4	d5	d6	d7	d8
MX17004	2010	1	tmax	—	—	—	—	—	—	—	—
MX17004	2010	1	tmin	—	—	—	—	—	—	—	—
MX17004	2010	2	tmax	—	27.3	24.1	—	—	—	—	—
MX17004	2010	2	tmin	—	14.4	14.4	—	—	—	—	—
MX17004	2010	3	tmax	—	—	—	—	32.1	—	—	—
MX17004	2010	3	tmin	—	—	—	—	14.2	—	—	—
MX17004	2010	4	tmax	—	—	—	—	—	—	—	—
MX17004	2010	4	tmin	—	—	—	—	—	—	—	—
MX17004	2010	5	tmax	—	—	—	—	—	—	—	—
MX17004	2010	5	tmin	—	—	—	—	—	—	—	—

Variable

Tidy Version

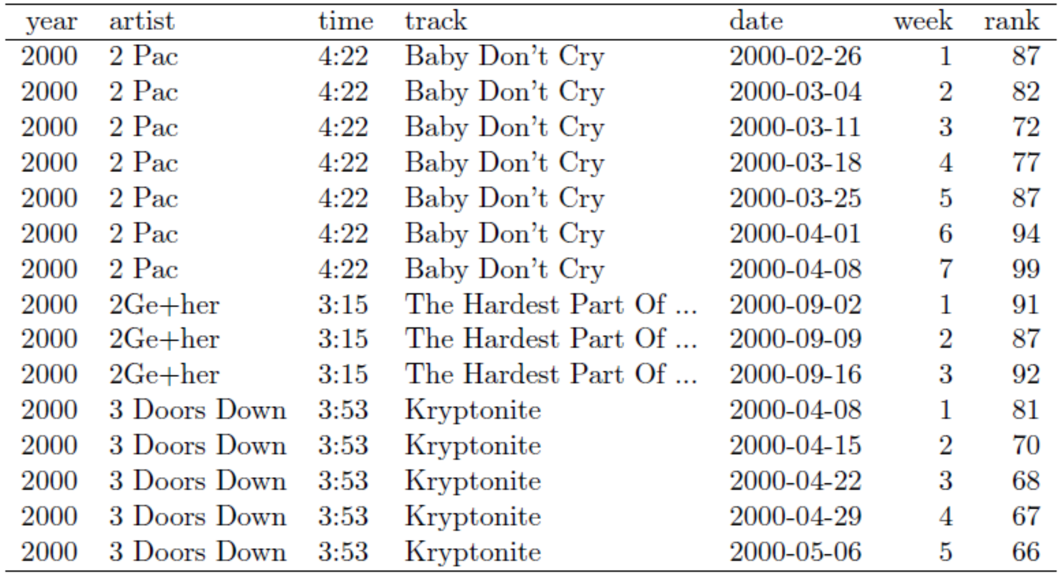


id	date	tmax	tmin
MX17004	2010-01-30	27.8	14.5
MX17004	2010-02-02	27.3	14.4
MX17004	2010-02-03	24.1	14.4
MX17004	2010-02-11	29.7	13.4
MX17004	2010-02-23	29.9	10.7
MX17004	2010-03-05	32.1	14.2
MX17004	2010-03-10	34.5	16.8
MX17004	2010-03-16	31.1	17.6
MX17004	2010-04-27	36.3	16.7
MX17004	2010-05-27	33.2	18.2

TIDY DATA: THEORY AND PRACTICE

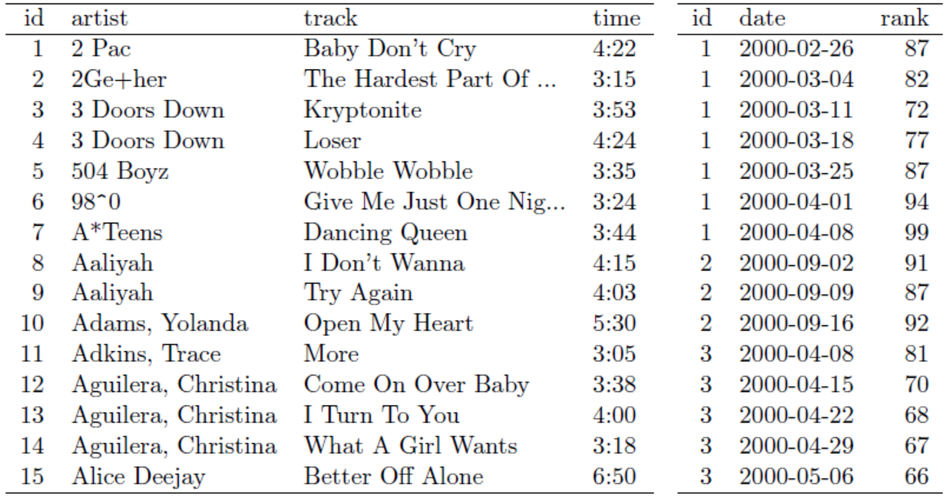
COMMON PROBLEM #4: Multiple Types of Observational Units are Stored in the Same Table

“Messy” Version



year	artist	time	track	date	week	rank
2000	2 Pac	4:22	Baby Don’t Cry	2000-02-26	1	87
2000	2 Pac	4:22	Baby Don’t Cry	2000-03-04	2	82
2000	2 Pac	4:22	Baby Don’t Cry	2000-03-11	3	72
2000	2 Pac	4:22	Baby Don’t Cry	2000-03-18	4	77
2000	2 Pac	4:22	Baby Don’t Cry	2000-03-25	5	87
2000	2 Pac	4:22	Baby Don’t Cry	2000-04-01	6	94
2000	2 Pac	4:22	Baby Don’t Cry	2000-04-08	7	99
2000	2Ge+her	3:15	The Hardest Part Of ...	2000-09-02	1	91
2000	2Ge+her	3:15	The Hardest Part Of ...	2000-09-09	2	87
2000	2Ge+her	3:15	The Hardest Part Of ...	2000-09-16	3	92
2000	3 Doors Down	3:53	Kryptonite	2000-04-08	1	81
2000	3 Doors Down	3:53	Kryptonite	2000-04-15	2	70
2000	3 Doors Down	3:53	Kryptonite	2000-04-22	3	68
2000	3 Doors Down	3:53	Kryptonite	2000-04-29	4	67
2000	3 Doors Down	3:53	Kryptonite	2000-05-06	5	66

“Tidy” Version



id	artist	track	time
1	2 Pac	Baby Don’t Cry	4:22
2	2Ge+her	The Hardest Part Of ...	3:15
3	3 Doors Down	Kryptonite	3:53
4	3 Doors Down	Loser	4:24
5	504 Boyz	Wobble Wobble	3:35
6	98^0	Give Me Just One Nig...	3:24
7	A*Teens	Dancing Queen	3:44
8	Aaliyah	I Don’t Wanna	4:15
9	Aaliyah	Try Again	4:03
10	Adams, Yolanda	Open My Heart	5:30
11	Adkins, Trace	More	3:05
12	Aguilera, Christina	Come On Over Baby	3:38
13	Aguilera, Christina	I Turn To You	4:00
14	Aguilera, Christina	What A Girl Wants	3:18
15	Alice Deejay	Better Off Alone	6:50

id	date	rank
1	2000-02-26	87
1	2000-03-04	82
1	2000-03-11	72
1	2000-03-18	77
1	2000-03-25	87
1	2000-04-01	94
1	2000-04-08	99
2	2000-09-02	91
2	2000-09-09	87
2	2000-09-16	92
3	2000-04-08	81
3	2000-04-15	70
3	2000-04-22	68
3	2000-04-29	67
3	2000-05-06	66

Normalization is useful for tidying and eliminating inconsistencies. However there are few data analysis tools that work directly with relational data, so analysis usually also requires denormalization or merging the datasets back into one table.

TIDY DATA: THEORY AND PRACTICE

COMMON PROBLEM #5: A Single Observational Unit is Stored in Multiple Tables

Messy Version

Year: 2018

Rank	Male name	Pct total males	Female name	Pct total females
1	Liam	1.03%	Emma	1.01%
2	Noah	0.95%	Olivia	0.97%
3	William	0.75%	Ava	0.81%
4	James	0.70%	Isabella	0.78%
5	Oliver	0.69%	Sophia	0.75%
6	Benjamin	0.69%	Charlotte	0.70%
7	Elijah	0.67%	Mia	0.68%
8	Lucas	0.65%	Amelia	0.67%
9	Mason	0.64%	Harper	0.57%
10	Logan	0.64%	Evelyn	0.56%

Year: 2019

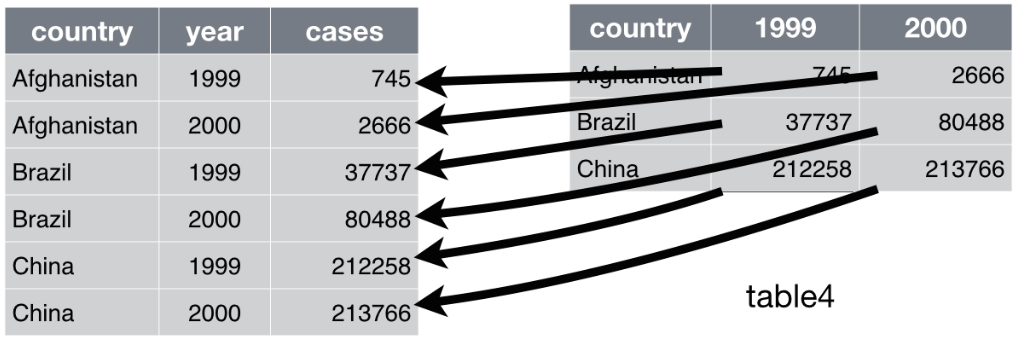
Rank	Male name	Pct total males	Female name	Pct total females
1	Liam	1.07%	Olivia	1.01%
2	Noah	1.00%	Emma	0.94%
3	Oliver	0.73%	Ava	0.79%
4	William	0.71%	Sophia	0.75%
5	Elijah	0.70%	Isabella	0.73%
6	James	0.69%	Charlotte	0.72%
7	Benjamin	0.68%	Amelia	0.71%
8	Lucas	0.65%	Mia	0.68%
9	Mason	0.60%	Harper	0.57%
10	Ethan	0.59%	Evelyn	0.57%

Tidy Version

Rank	Year	Male name	Pct total males	Female name	Pct total females
1	2018	Liam	1.03%	Emma	1.01%
2	2018	Noah	0.95%	Olivia	0.97%
3	2018	William	0.75%	Ava	0.81%
4	2018	James	0.70%	Isabella	0.78%
5	2018	Oliver	0.69%	Sophia	0.75%
6	2018	Benjamin	0.69%	Charlotte	0.70%
7	2018	Elijah	0.67%	Mia	0.68%
8	2018	Lucas	0.65%	Amelia	0.67%
9	2018	Mason	0.64%	Harper	0.57%
10	2018	Logan	0.64%	Evelyn	0.56%
1	2019	Liam	1.07%	Olivia	1.01%
2	2019	Noah	1.00%	Emma	0.94%
3	2019	Oliver	0.73%	Ava	0.79%
4	2019	William	0.71%	Sophia	0.75%
5	2019	Elijah	0.70%	Isabella	0.73%
6	2019	James	0.69%	Charlotte	0.72%
7	2019	Benjamin	0.68%	Amelia	0.71%
8	2019	Lucas	0.65%	Mia	0.68%
9	2019	Mason	0.60%	Harper	0.57%
10	2019	Ethan	0.59%	Evelyn	0.57%

TIDY DATA: TOOLS

Tool	Description	Tidyverse Data Sample	Syntax
Pivot Longer (Taller)	Column names are not names of variables, but values of a variable	table4a	pivot_longer(data, columns = c(columns), names_to = “new name for columns”, values_to = “new name for values”)

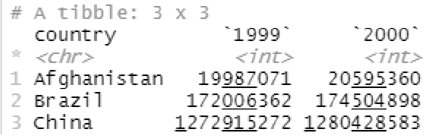


country	year	cases
Afghanistan	1999	745
Afghanistan	2000	2666
Brazil	1999	37737
Brazil	2000	80488
China	1999	212258
China	2000	213766

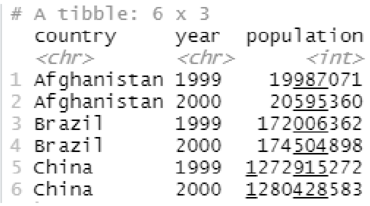
country	1999	2000
Afghanistan	745	2666
Brazil	37737	80488
China	212258	213766

table4

```
table4a
table4a %>%
  pivot_longer(c(`1999`, `2000`),
               names_to = "year",
               values_to = “cases")
```



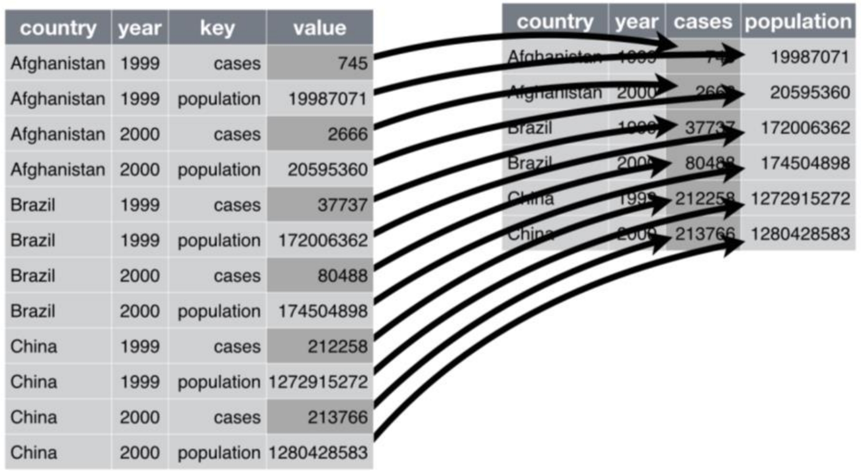
```
# A tibble: 3 x 3
  country          `1999`     `2000`
* <chr>             <int>      <int>
1 Afghanistan    19987071   20595360
2 Brazil        172006362  174504898
3 China        1272915272 1280428583
```



```
# A tibble: 6 x 3
  country     year  population
  <chr>       <chr>      <int>
1 Afghanistan 1999    19987071
2 Afghanistan 2000    20595360
3 Brazil      1999   172006362
4 Brazil      2000   174504898
5 China       1999  1272915272
6 China       2000  1280428583
```

TIDY DATA: TOOLS

Tool	Description	Tidyverse Data Sample	Syntax
Pivot Wider (Wider)	An observation is scattered across multiple rows.	table2	pivot_wider(data, names_from = [source of new column names], values_from = [source of new column values])

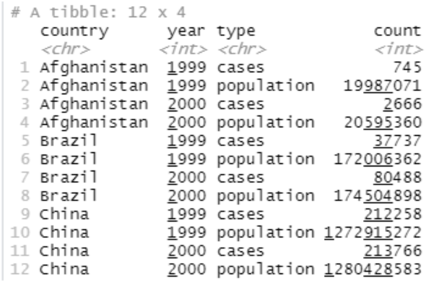


country	year	key	value
Afghanistan	1999	cases	745
Afghanistan	1999	population	19987071
Afghanistan	2000	cases	2666
Afghanistan	2000	population	20595360
Brazil	1999	cases	37737
Brazil	1999	population	172006362
Brazil	2000	cases	80488
Brazil	2000	population	174504898
China	1999	cases	212258
China	1999	population	1272915272
China	2000	cases	213766
China	2000	population	1280428583

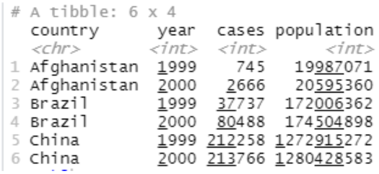
country	year	cases	population
Afghanistan	1999	745	19987071
Afghanistan	2000	2666	20595360
Brazil	1999	37737	172006362
Brazil	2000	80488	174504898
China	1999	212258	1272915272
China	2000	213766	1280428583

table2

```
table2
table2 %>%
  pivot_wider(names_from = type,
              values_from = count)
```



```
# A tibble: 12 x 4
   country      year type            count
   <chr>       <int> <chr>           <int>
 1 Afghanistan  1999 cases             745
 2 Afghanistan  1999 population   19987071
 3 Afghanistan  2000 cases            2666
 4 Afghanistan  2000 population   20595360
 5 Brazil       1999 cases           37737
 6 Brazil       1999 population  172006362
 7 Brazil       2000 cases           80488
 8 Brazil       2000 population  174504898
 9 China        1999 cases          212258
10 China        1999 population 1272915272
11 China        2000 cases          213766
12 China        2000 population 1280428583
```

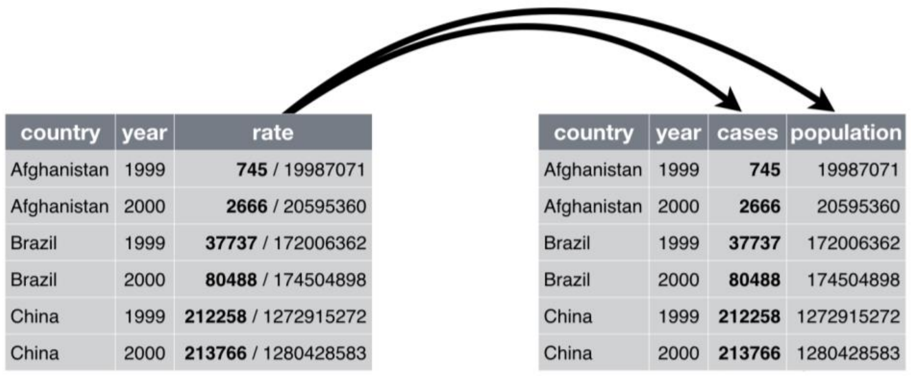


```
# A tibble: 6 x 4
  country      year  cases population
  <chr>       <int>  <int>      <int>
1 Afghanistan  1999    745   19987071
2 Afghanistan  2000   2666   20595360
3 Brazil       1999  37737  172006362
4 Brazil       2000  80488  174504898
5 China        1999 212258 1272915272
6 China        2000 213766 1280428583
```

TIDY DATA: TOOLS

Tool	Description	Tidyverse Data Sample	Syntax

Separate	Pull apart one column into multiple columns	table3	separate(data,
			col = [column to separate],
			into = c(“names of new columns”),
			sep = “separator character”)

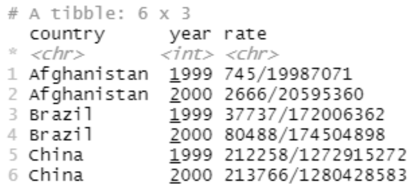


country	year	rate
Afghanistan	1999	**745** / 19987071
Afghanistan	2000	**2666** / 20595360
Brazil	1999	**37737** / 172006362
Brazil	2000	**80488** / 174504898
China	1999	**212258** / 1272915272
China	2000	**213766** / 1280428583

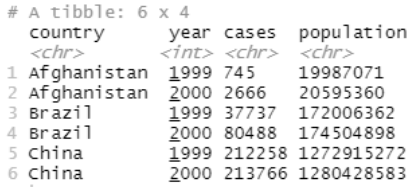
country	year	cases	population
Afghanistan	1999	**745**	19987071
Afghanistan	2000	**2666**	20595360
Brazil	1999	**37737**	172006362
Brazil	2000	**80488**	174504898
China	1999	**212258**	1272915272
China	2000	**213766**	1280428583

table3

```
table3
table3 %>%
  separate(rate,
              into = c(“cases”, “population”),
              sep = “/”)
```



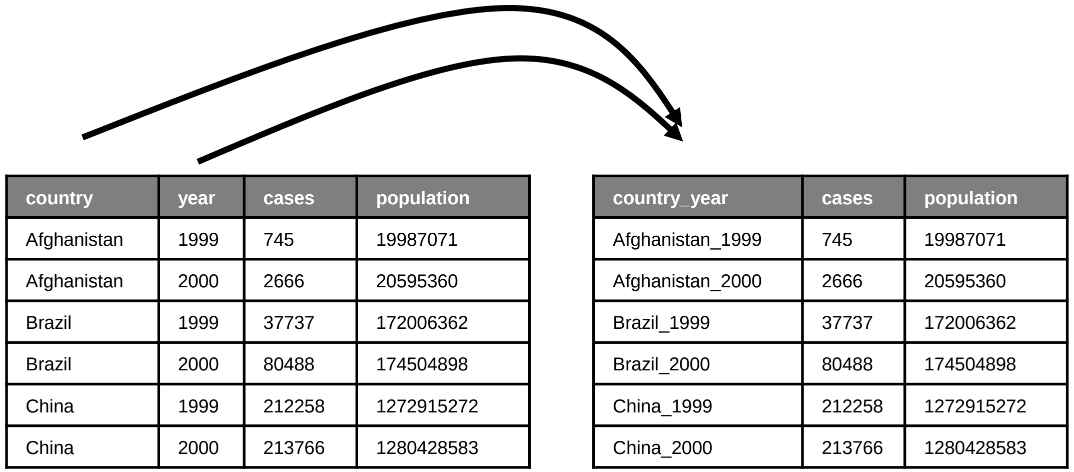
```
# A tibble: 6 x 3
  country      year rate
* <chr>       <int> <chr>
1 Afghanistan  1999 745/19987071
2 Afghanistan  2000 2666/20595360
3 Brazil       1999 37737/172006362
4 Brazil       2000 80488/174504898
5 China        1999 212258/1272915272
6 China        2000 213766/1280428583
```



```
# A tibble: 6 x 4
  country      year cases  population
  <chr>       <int> <chr>  <chr>
1 Afghanistan  1999 745    19987071
2 Afghanistan  2000 2666   20595360
3 Brazil       1999 37737  172006362
4 Brazil       2000 80488  174504898
5 China        1999 212258 1272915272
6 China        2000 213766 1280428583
```

TIDY DATA: TOOLS

Tool	Description	Tidyverse Data Sample	Syntax
Unite	Combines multiple columns into a single column	table1	unite(data, col =“name for new united column”, … = [the columns you want to unite], sep = “separator character with _ as default”)

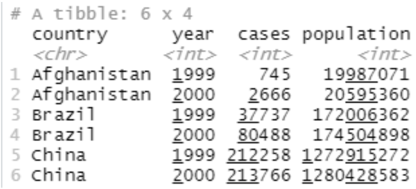


country	year	cases	population
Afghanistan	1999	745	19987071
Afghanistan	2000	2666	20595360
Brazil	1999	37737	172006362
Brazil	2000	80488	174504898
China	1999	212258	1272915272
China	2000	213766	1280428583

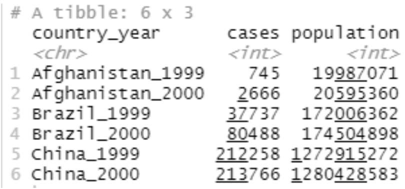
country_year	cases	population
Afghanistan_1999	745	19987071
Afghanistan_2000	2666	20595360
Brazil_1999	37737	172006362
Brazil_2000	80488	174504898
China_1999	212258	1272915272
China_2000	213766	1280428583

table1

```
table1
table1 %>%
  unite(col = “country_year”,
          country,
          year)
```

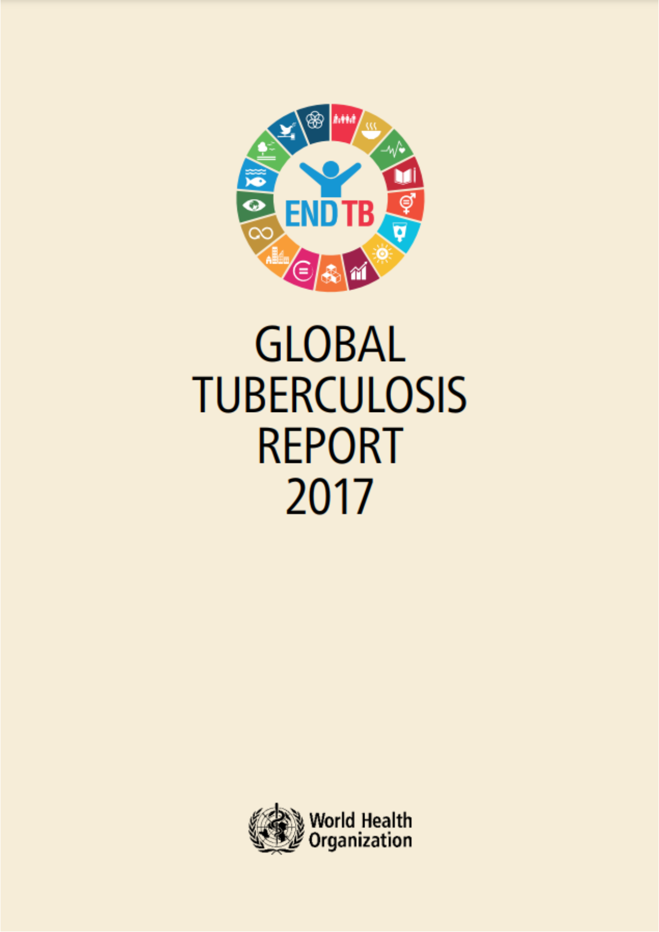


```
# A tibble: 6 x 4
  country      year  cases population
  <chr>       <int>  <int>      <int>
1 Afghanistan  1999    745   19987071
2 Afghanistan  2000   2666   20595360
3 Brazil       1999  37737  172006362
4 Brazil       2000  80488  174504898
5 China        1999 212258 1272915272
6 China        2000 213766 1280428583
```

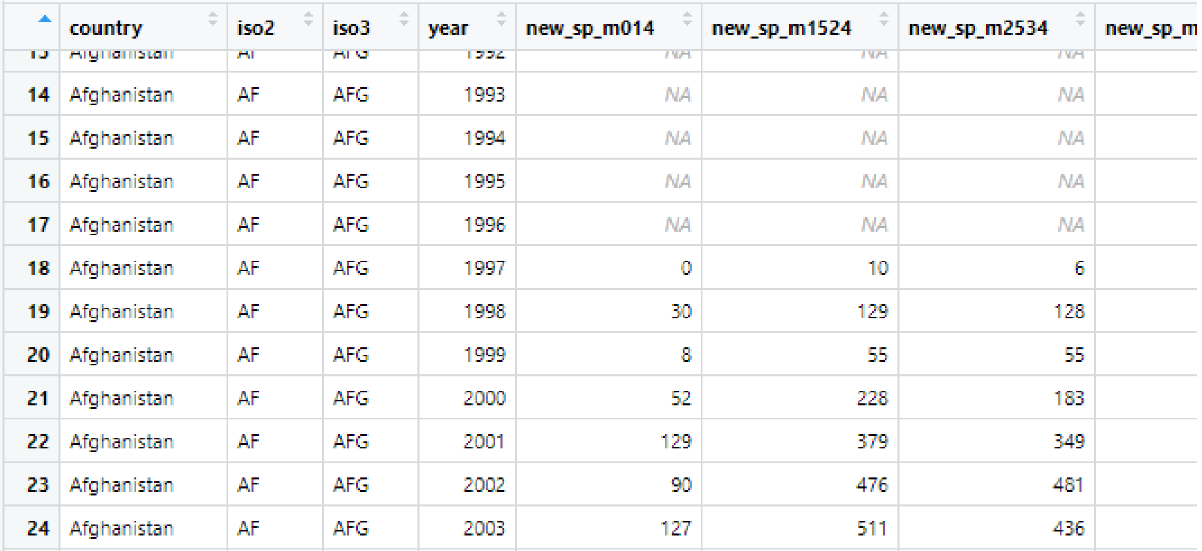


```
# A tibble: 6 x 3
  country_year      cases population
  <chr>             <int>      <int>
1 Afghanistan_1999    745   19987071
2 Afghanistan_2000   2666   20595360
3 Brazil_1999       37737  172006362
4 Brazil_2000       80488  174504898
5 China_1999       212258 1272915272
6 China_2000       213766 1280428583
```

TIDY DATA: CASE STUDY



```
who %>% View()
```



	country	iso2	iso3	year	new_sp_m014	new_sp_m1524	new_sp_m2534	new_sp_m
13	Afghanistan	AF	AFG	1992	NA	NA	NA	
14	Afghanistan	AF	AFG	1993	NA	NA	NA	
15	Afghanistan	AF	AFG	1994	NA	NA	NA	
16	Afghanistan	AF	AFG	1995	NA	NA	NA	
17	Afghanistan	AF	AFG	1996	NA	NA	NA	
18	Afghanistan	AF	AFG	1997	0	10	6	
19	Afghanistan	AF	AFG	1998	30	129	128	
20	Afghanistan	AF	AFG	1999	8	55	55	
21	Afghanistan	AF	AFG	2000	52	228	183	
22	Afghanistan	AF	AFG	2001	129	379	349	
23	Afghanistan	AF	AFG	2002	90	476	481	
24	Afghanistan	AF	AFG	2003	127	511	436	

Info Encoded in Column Headers

New (if the TB cases are new or old; all of these are new)

Type of TB (rel, ep, sn, sp)

Patient Sex (m, f)

Age Group (014 = 0 to 14 years, etc.)

NEXT WEEK…

- Case Study Showcase
- Chapter 13 – Relational Data

GETTING HELP

- Ask questions during our call
- Google
- Stack Overflow
- Slack
- Office Hours r4ds.io/calendar
- Twitter #rstats
- r4ds answer keys: Jeff Arnold (preferred) or Bryan Shalloway (also good)
- Cheatsheets

RStudio®