

Tidy data

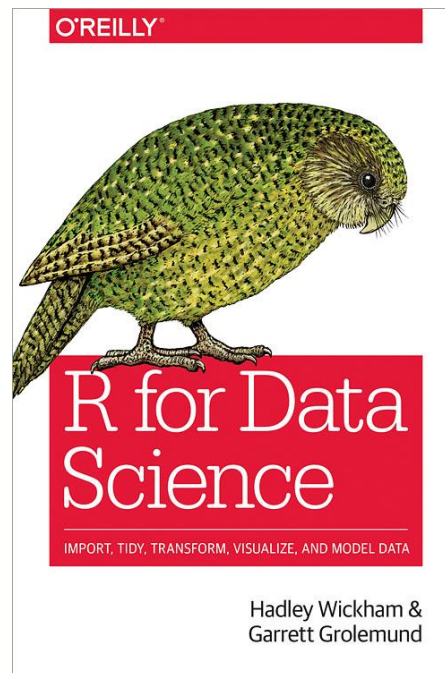
Tidy data

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

— Hadley Wickham

Acknowledgements

Material for this Tidy data and Relational data sessions draws heavily on Chapters 9 and 10 of:



Tidy data

country	year	cases	population
Afghanistan	1999	745	19997071
Afghanistan	2000	7666	200095360
Brazil	1999	3737	172006362
Brazil	2000	80488	174004898
China	1999	212258	1272015272
China	2000	210766	128042583

variables

country	year	cases	population
Afghanistan	1999	745	19997071
Afghanistan	2000	7666	200095360
Brazil	1999	3737	172006362
Brazil	2000	80488	174004898
China	1999	212258	1272015272
China	2000	210766	128042583

observations

country	year	cases	population
Afghanistan	99	745	19997071
Afghanistan	00	7666	200095360
Brazil	99	3737	172006362
Brazil	00	80488	174004898
China	99	212258	1272015272
China	00	210766	128042583

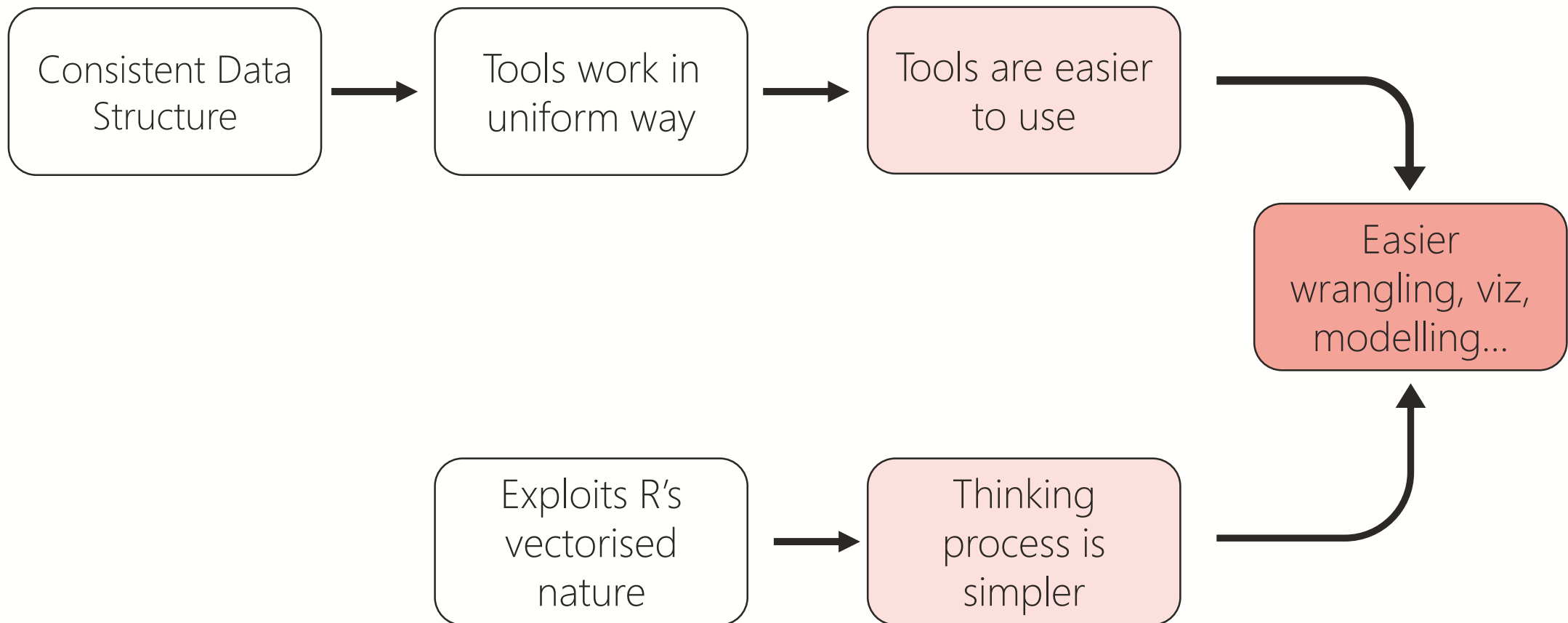
values

Each variable has its own column.

Each observation has its own row.

Each value has its own cell.

Why tidy data?



Tidy data

Load the original `tb_cases` and population data sets (as you did yesterday afternoon. This (wide table) is similar to the format used by the W.H.O.

Your turn

Having loaded these tables, can we now join them two tables together?

Tidying data

1999	2000	2001

colnames are
year *values*

number of
cases

gather

```
tb_cases %>%
```

```
  gather(year, cases, )
```

1999	2000	2001

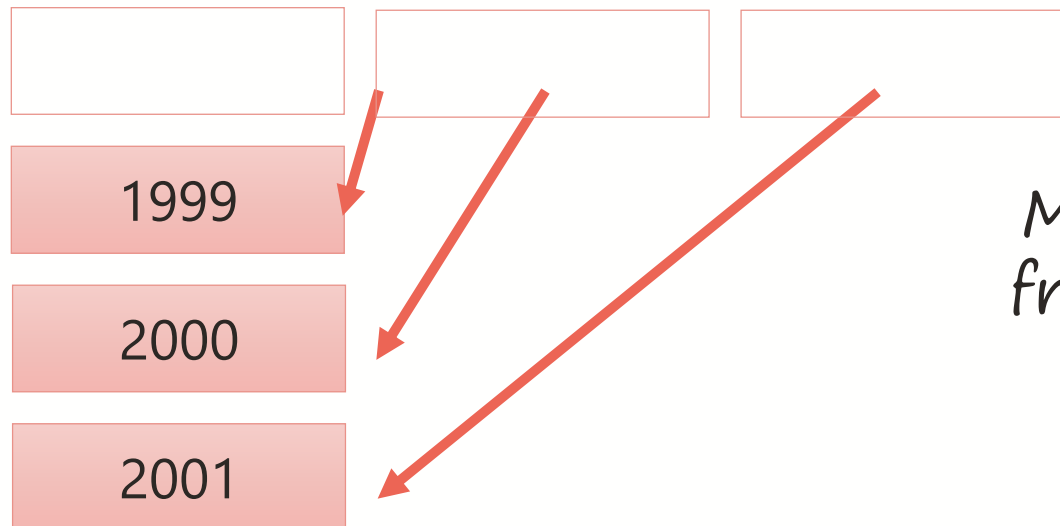
← colnames are
year *values*

← number of
cases

gather

```
tb_cases %>%
```

```
  gather(year, cases, )
```



*Move values
from header
to cells*

gather

```
tb_cases %>%
```

```
gather(year, cases, )
```

name



--

1999

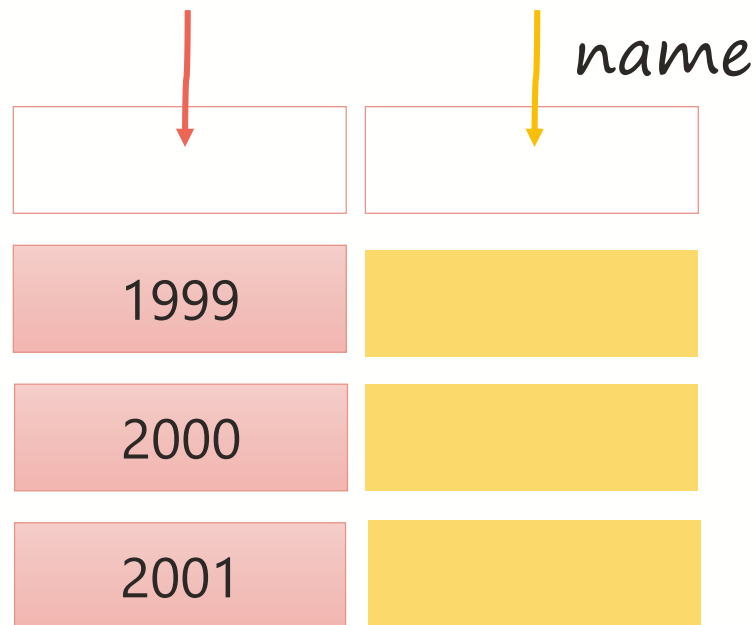
2000

2001

gather

```
tb_cases %>%
```

```
gather(year, cases, )
```



gather

tb_cases %>%

gather(year, cases, 2:4)

The diagram illustrates the effect of the `gather` function. It shows a transformation from three columns (2, 3, 4) into a single column named 'year' and another column named 'cases'. The 'year' column contains the values 1999, 2000, and 2001. The 'cases' column is represented by three empty yellow boxes, indicating that the data from the original columns 3 and 4 is being gathered into this single column.

1999		
2000		
2001		

Original table

The diagram shows the original table structure. It has three columns labeled 2, 3, and 4. Column 2 contains the years 1999, 2000, and 2001. Columns 3 and 4 are represented by empty yellow boxes, indicating that the data from these columns is being gathered into a single column named 'cases'.

2	3	4
1999		

Note:
You will actually
need 2:5

Your turn: Tidy “pop”

pop %>%

gather(, ,)

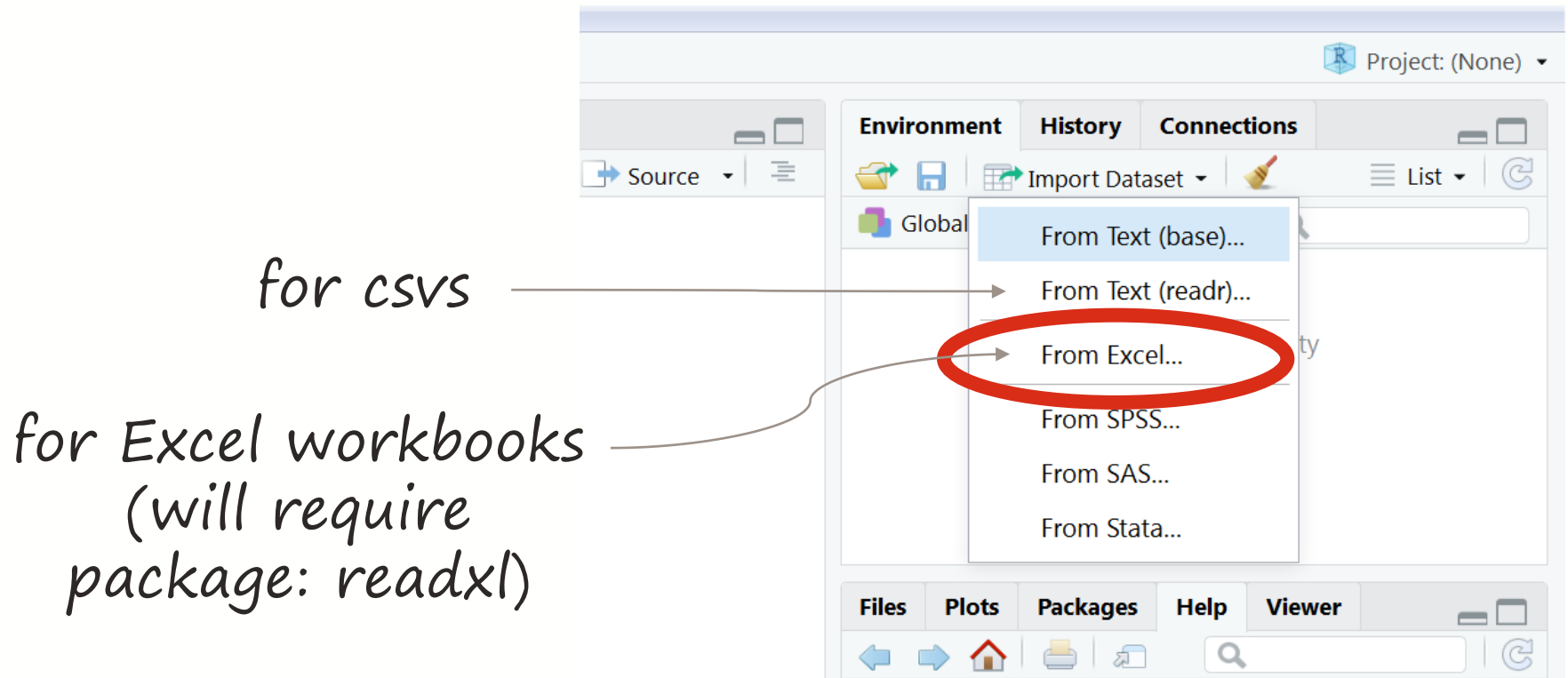
1999	2000	2001

← colnames are
year *values*

← population
values

Import Excel workbooks

Import `data_spread.xlsx` . Assign to object "table2"




spread

does the opposite of gather:


```
table2 %>%
```

```
  spread(key = type, value = count)
```

Column
containing
variable
names



Column that
contains
values



Real life example:

You've found ONS population projections that you'd like to use for forecasting future healthcare utilisation. Load into R and tidy!

This work is licensed as

Creative Commons

Attribution-ShareAlike 4.0

International

To view a copy of this license, visit

<https://creativecommons.org/licenses/by-sa/4.0/>

End

Tidy data?

```
# A tibble: 6 x 6
  manufacturer model      displ  year   cty   hwy
  <chr>         <chr>    <dbl> <int> <int> <int>
1 audi         a4 quattro    2.00  2008   19    27
2 dodge        durango 4wd    3.90  1999   13    17
3 dodge        ram 1500 pickup 4wd  4.70  2008   12    16
4 ford         f150 pickup 4wd    4.60  2008   13    17
5 nissan        pathfinder 4wd    3.30  1999   15    17
6 subaru        forester awd    2.50  2008   18    23
```

Why tidy data?

```
# A tibble: 12 x 6
```

	manufacturer	model	displ	year	environment	mpg
	<chr>	<chr>	<dbl>	<int>	<chr>	<int>
1	audi	a4 quattro	2.00	2008	cty	19
2	audi	a4 quattro	2.00	2008	hwy	27
3	dodge	durango 4wd	3.90	1999	cty	13
4	dodge	ram 1500 pickup 4wd	4.70	2008	cty	12
5	dodge	durango 4wd	3.90	1999	hwy	17
6	dodge	ram 1500 pickup 4wd	4.70	2008	hwy	16
7	ford	f150 pickup 4wd	4.60	2008	cty	13
8	ford	f150 pickup 4wd	4.60	2008	hwy	17