Data Science for Everyone – Data Wrangling – Tidy Data

Dr. Ab Mosca (they/them)

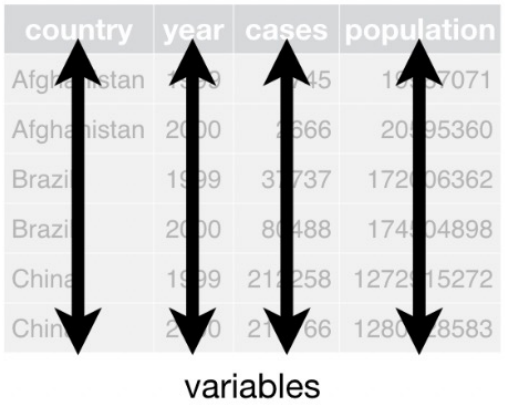
Slides based off slides courtesy of Jordan Crouser (https://jcrouser.github.io/)

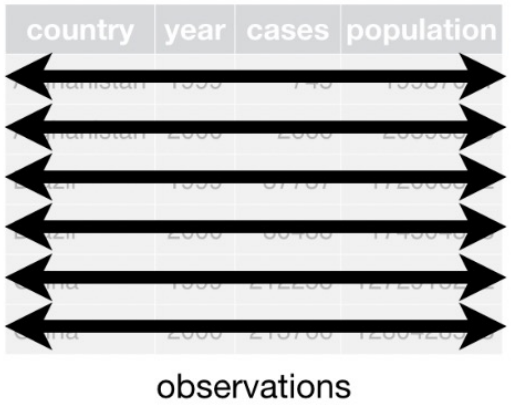
Plan for Today

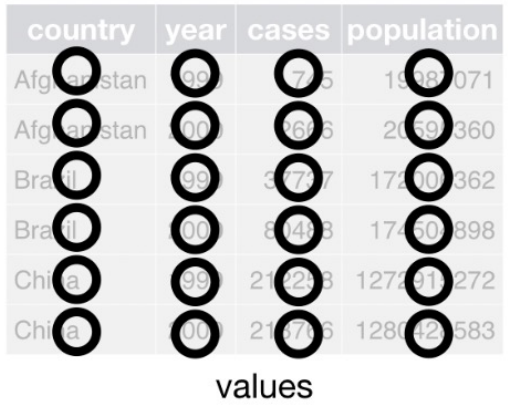
- Clean messy data with R

Reminder: Table Vocabulary

- When data is tidy, every column is a variable, every row is an observation, and every value has its own cell





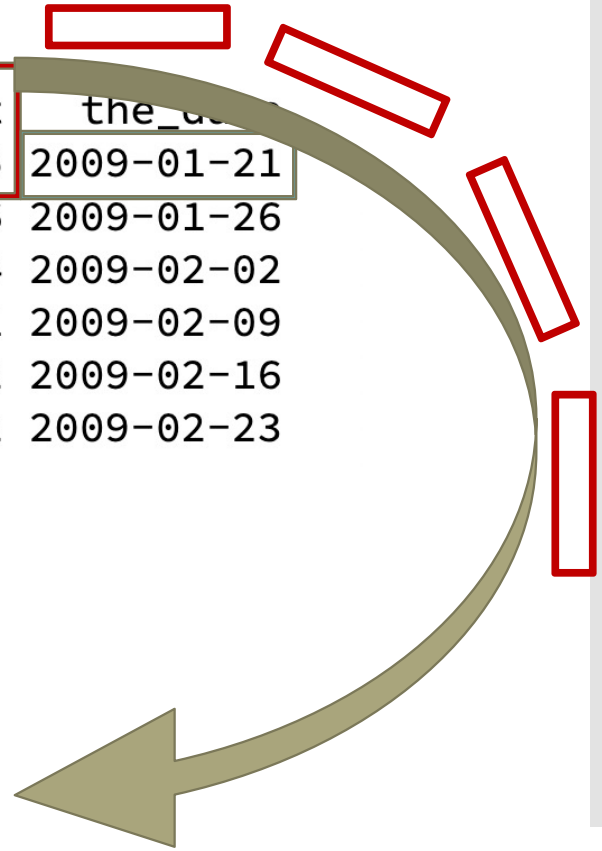


Tidy Data

- `pivot_longer`
  - Each observation gets its own row
  - Number of rows increases (table gets longer)

```
##   Republican Independent Democrat   the_date
## 1         16          47        85 2009-01-21
## 2         18          48        86 2009-01-26
## 3         17          45        84 2009-02-02
## 4         18          46        81 2009-02-09
## 5         17          46        82 2009-02-16
## 6         18          44        82 2009-02-23
```

```
## # A tibble: 4 x 3
##   the_date   party       approval
##   <date>     <chr>          <int>
## 1 2009-01-21 Republican        16
## 2 2009-01-21 Independent       47
## 3 2009-01-21 Democrat          85
## 4 2009-01-26 Republican        18
```

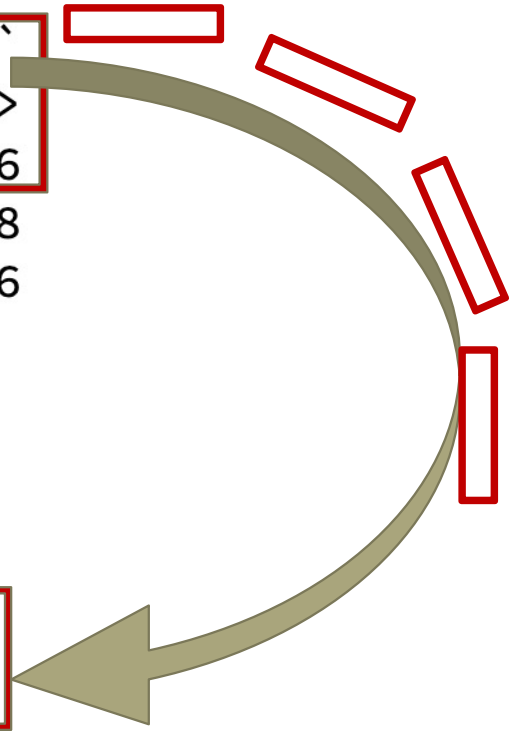


Tidy Data

- `pivot_longer`
  - Each observation gets its own row
  - Number of rows increases (table gets longer)

```
## # A tibble: 3 x 3
##   country     1999`  `2000`
## * <chr>       <int>   <int>
## 1 Afghanistan   745    2666
## 2 Brazil      37737   80488
## 3 China      212258  213766
```

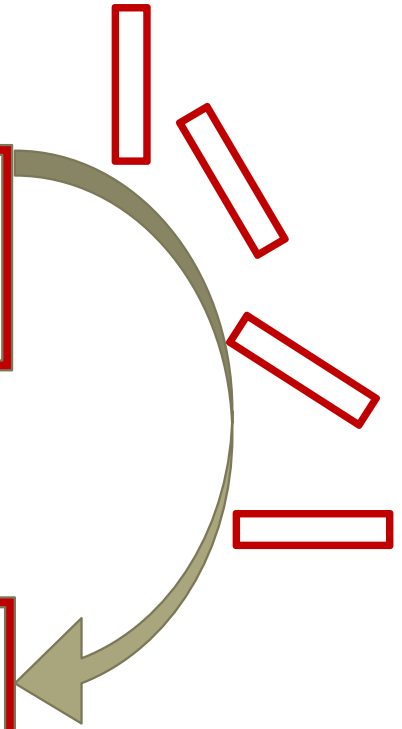
```
## # A tibble: 6 x 3
##   country     year   cases
##   <chr>       <chr>  <int>
## 1 Afghanistan 1999     745
## 2 Afghanistan 2000    2666
## 3 Brazil      1999   37737
## 4 Brazil      2000   80488
## 5 China       1999  212258
```



Tidy Data

- `pivot_wider`
  - Each observation gets its own row
  - Number of rows decreases

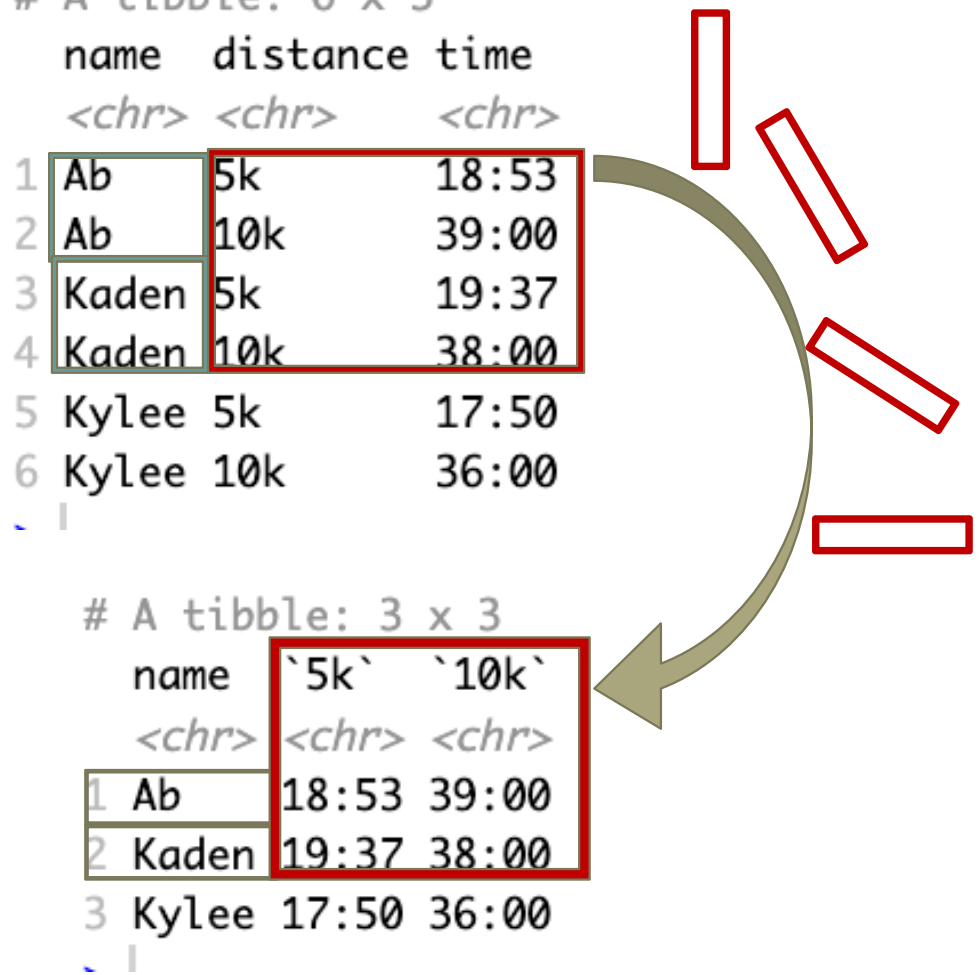
```
## # A tibble: 6 x 4
##   country      year type            count
##   <chr>       <int> <chr>           <int>
## 1 Afghanistan  1999 cases             745
## 2 Afghanistan  1999 population  19987071
## 3 Afghanistan  2000 cases            2666
## 4 Afghanistan  2000 population  20595360
## 5 Brazil       1999 cases            37737
## 6 Brazil       1999 population 172006362
```



```
## # 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

- `pivot_wider`
  - Each observation gets its own row
  - Number of rows decreases



```
# A tibble: 6 x 3
  name   distance time
  <chr>  <chr>    <chr>
1 Ab     5k       18:53
2 Ab     10k      39:00
3 Kaden  5k       19:37
4 Kaden  10k      38:00
5 Kylee  5k       17:50
6 Kylee  10k      36:00
```

```
# A tibble: 3 x 3
  name   `5k`  `10k`
  <chr>  <chr> <chr>
1 Ab     18:53 39:00
2 Kaden  19:37 38:00
3 Kylee  17:50 36:00
```

- `tidyr`
  - R package that helps make data tidy
  - We will primarily use two functions:
    - `pivot_longer()`
    - `pivot_wider()`

pivot_longer()

- Used when column headers contain data
- Increases number of rows


```
table %>%
   pivot_longer(<value-columns>, names_to = , values_to = )
```

Set of columns with values

New column name for column data

New column name for cell data

pivot_longer()

- Used when column headers contain data
- Increases number of rows

```
## # A tibble: 3 x 3
##   country     1999`  `2000`
## * <chr>       <int>   <int>
## 1 Afghanistan   745    2666
## 2 Brazil      37737   80488
## 3 China      212258  213766
```

```
## # A tibble: 6 x 3
##   country     year   cases
##   <chr>       <chr>  <int>
## 1 Afghanistan 1999     745
## 6 China                3766
```

Set of columns with values

New column name for column data

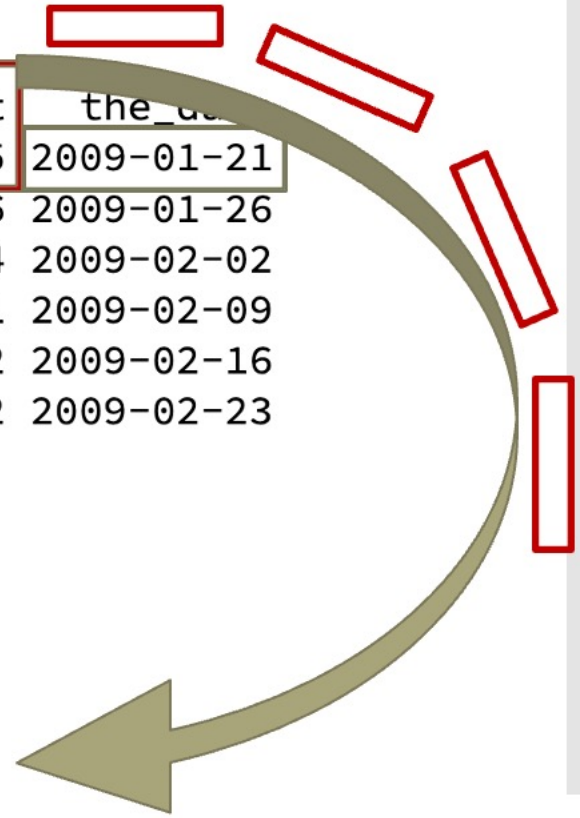
New column name for cell data

```
table4a %>%
   pivot_longer(-country,
                names_to = "year",
                values_to = "cases")
```

pivot_longer()

What would the code for this pivot be?

```
##   Republican Independent Democrat   the_da
## 1         16          47        85 2009-01-21
## 2         18          48        86 2009-01-26
## 3         17          45        84 2009-02-02
## 4         18          46        81 2009-02-09
## 5         17          46        82 2009-02-16
## 6         18          44        82 2009-02-23
```



```
## # A tibble: 4 x 3
##   the_date   party       approval
##   <date>     <chr>          <int>
## 1 2009-01-21 Republican        16
## 2 2009-01-21 Independent       47
## 3 2009-01-21 Democrat          85
## 4 2009-01-26 Republican        18
```

pivot_longer()

What would the code for this pivot be?

```
##   Republican Independent Democrat   the_date
## 1         16          47        85 2009-01-21
## 2         18          48        86 2009-01-26
## 3         17          45        84 2009-02-02
## 4         18          46        81 2009-02-09
## 5         17          46        82 2009-02-16
## 6         18          44        82 2009-02-23
```

```
## # A tibble: 4 x 3
##   the_date   party       approval
##   <date>     <chr>          <int>
## 1 2009-01-21 Republican        16
## 2 2009-01-21 Independent       47
## 3 2009-01-21 Democrat          85
## 4 2009-01-26 Republican        18
```

Set of columns with values

New column name for column data

New column name for cell data

```
table %>% pivot_longer(-the_date,
                       names_to = "party",
                       values_to = "approval")
```

`pivot_wider()`

- Used when observations are split between rows
- Decreases number of rows

```
table %>%
   pivot_wider(names_from =, values_from = )
```

Column to take new column names from

Column to take values from

pivot_wider()

- Used when observations are split between rows
- Decreases number of rows

```
## # A tibble: 6 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
```

```
## # 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
```

Column to take new column names from

Column to take values from

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

`pivot_wider()`

...veen rows

What would the code for this pivot be?

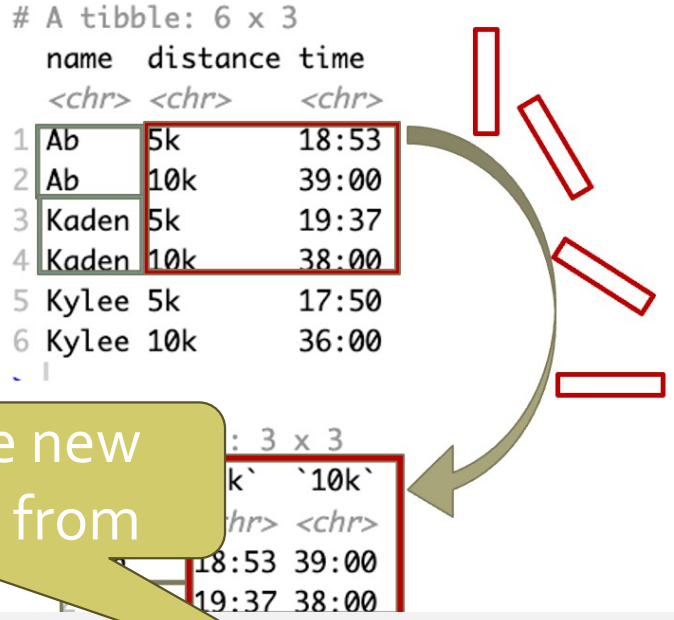
```
# A tibble: 6 x 3
  name  distance time
  <chr> <chr>    <chr>
1 Ab    5k       18:53
2 Ab    10k      39:00
3 Kaden 5k       19:37
4 Kaden 10k      38:00
5 Kylee 5k       17:50
6 Kylee 10k      36:00
```

```
# A tibble: 3 x 3
  name  `5k`  `10k`
  <chr> <chr> <chr>
1 Ab    18:53 39:00
2 Kaden 19:37 38:00
3 Kylee 17:50 36:00
```

pivot_wider()

veen rows

What would the code for this pivot be?



```
# A tibble: 6 x 3
  name  distance time
  <chr> <chr>    <chr>
1 Ab    5k       18:53
2 Ab    10k      39:00
3 Kaden 5k       19:37
4 Kaden 10k      38:00
5 Kylee 5k       17:50
6 Kylee 10k      36:00
```

```
: 3 x 3
k`   `10k`
hr> <chr>
18:53 39:00
19:37 38:00
```

Column to take new column names from

```
table %>% pivot_wider(names_from = distance,
                      values_from = time)
```

Column to take values from

Example

Load the hiv_deaths dataset from http://www.gapminder.org/data/

```
96
97 ```{r}
98 hiv <- read_csv("~/Downloads/hiv_deaths_in_children_1_59_months_total_deaths.csv")
99 ```
100
```

What are the dimensions of this dataset? Is it tidy?

Example

Load the hiv_deaths dataset from http://www.gapminder.org/data/

```
96
97 ```{r}
98 hiv <- read_csv("~/Downloads/hiv_deaths_in_children_1_59_months_total_deaths.csv")
99 ```
100
```

What are the dimensions of this dataset? Is it tidy?

```
105 ```{r}
106 head(hiv)
107 dim(hiv)
108 ```
```

A tibble: 6 x 31

country <chr>	1989 <chr>	1990 <chr>	1991 <chr>	1992 <chr>	1 <chr>
Afghanistan	10	11.6	13.3	15.6	1
Angola	64.4	95.1	136	190	2
Albania	0.21	0.24	0.28	0.27	0
Andorra	0.04	0.04	0.04	0.04	0
United Arab Emirates	0.53	0.61	0.68	0.76	0
Argentina	7.59	7.56	7.56	7.31	6

6 rows | 1-10 of 31 columns

```
[1] 204  31
```

Example

- Fix a data error

```
118     {}
119   hiv["2018"] <- as.character(hiv["2018"])
120
```

- Pivot the data table so that it is tidy

Example

- Fix a data error

```
119 hiv["2018"] <- as.character(hiv["2018"])
120
```

- Pivot the data table so that it is tidy

```
121 hiv <- hiv %>%
122   pivot_longer(-country,
123                names_to = "year",
124                values_to = "est_prevalence") %>%
125   mutate(year = sub("X", "", year))
126 ```
```

Example

- Now that your data is tidy, think of an interesting question, and answer it with a graph