MATH 118: Notes J

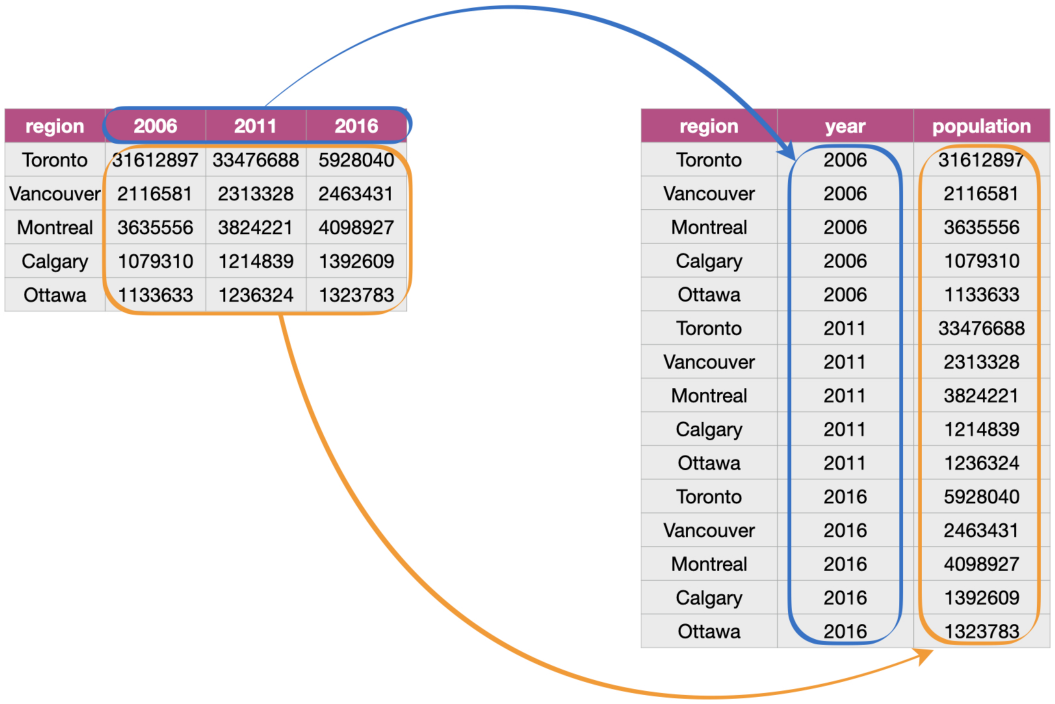
Pivoting

Reshaping with Pivoting – Why?

Data frames are often described as wide or long.

Wide when a row has more than one observation, and the units of observation are on one row each

Long when a row has only one observation, but the units of observation are repeated down the column



region	2006	2011	2016
Toronto	31612897	33476688	5928040
Vancouver	2116581	2313328	2463431
Montreal	3635556	3824221	4098927
Calgary	1079310	1214839	1392609
Ottawa	1133633	1236324	1323783

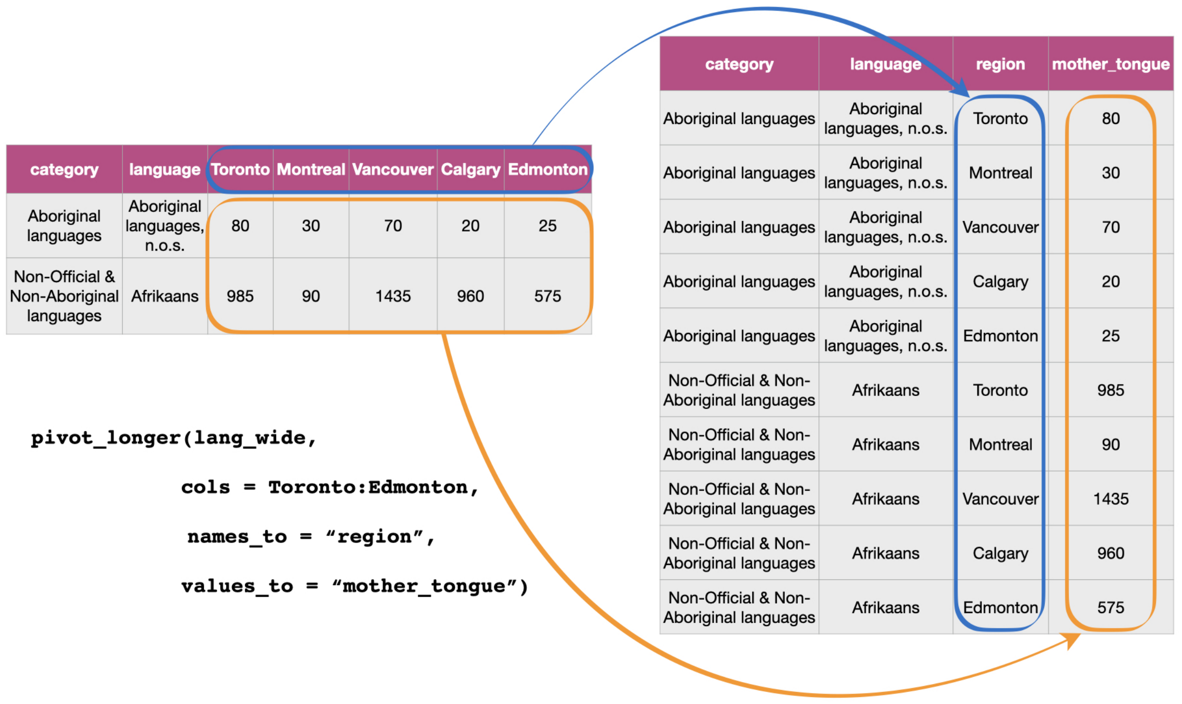
region	year	population
Toronto	2006	31612897
Vancouver	2006	2116581
Montreal	2006	3635556
Calgary	2006	1079310
Ottawa	2006	1133633
Toronto	2011	33476688
Vancouver	2011	2313328
Montreal	2011	3824221
Calgary	2011	1214839
Ottawa	2011	1236324
Toronto	2016	5928040
Vancouver	2016	2463431
Montreal	2016	4098927
Calgary	2016	1392609
Ottawa	2016	1323783

`canlang` dataset

```
#LOAD PACKAGES
library(tidyverse)

#LOAD DATA
lang_wide <- read.csv("https://raw.githubusercontent.com/UBC-DSCI/introduction-to-datascience/master/da
```

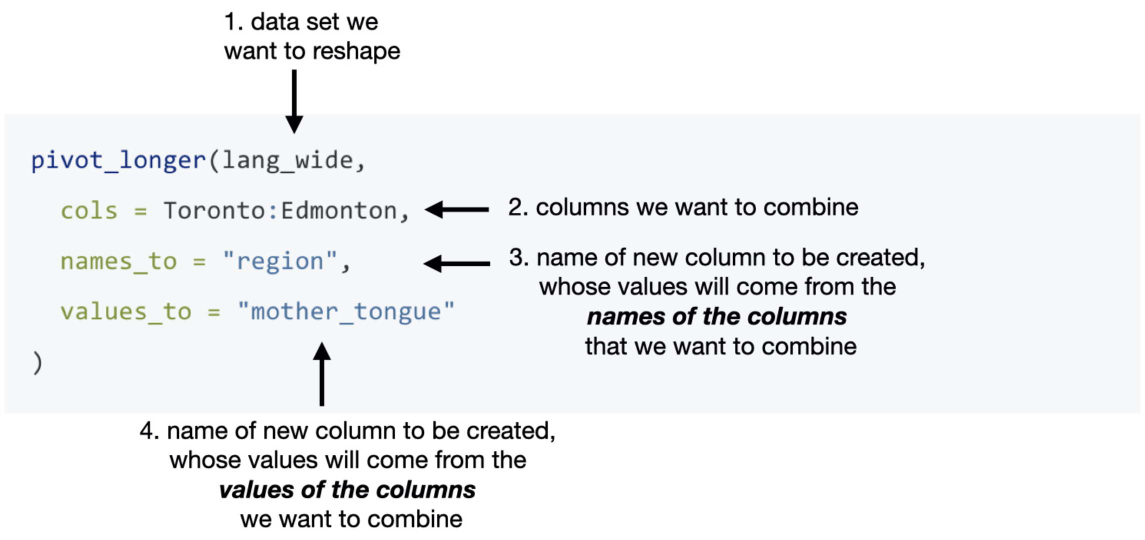
Pivot Longer



category	language	Toronto	Montreal	Vancouver	Calgary	Edmonton
Aboriginal languages	Aboriginal languages, n.o.s.	80	30	70	20	25
Non-Official & Non-Aboriginal languages	Afrikaans	985	90	1435	960	575

```
pivot_longer(lang_wide,
             cols = Toronto:Edmonton,
             names_to = “region”,
             values_to = “mother_tongue”)
```

category	language	region	mother_tongue
Aboriginal languages	Aboriginal languages, n.o.s.	Toronto	80
Aboriginal languages	Aboriginal languages, n.o.s.	Montreal	30
Aboriginal languages	Aboriginal languages, n.o.s.	Vancouver	70
Aboriginal languages	Aboriginal languages, n.o.s.	Calgary	20
Aboriginal languages	Aboriginal languages, n.o.s.	Edmonton	25
Non-Official & Non-Aboriginal languages	Afrikaans	Toronto	985
Non-Official & Non-Aboriginal languages	Afrikaans	Montreal	90
Non-Official & Non-Aboriginal languages	Afrikaans	Vancouver	1435
Non-Official & Non-Aboriginal languages	Afrikaans	Calgary	960
Non-Official & Non-Aboriginal languages	Afrikaans	Edmonton	575



```
pivot_longer(lang_wide,
  cols = Toronto:Edmonton,
  names_to = "region",
  values_to = "mother_tongue"
)
```

1. data set we want to reshape
2. columns we want to combine
3. name of new column to be created, whose values will come from the ***names of the columns*** that we want to combine
4. name of new column to be created, whose values will come from the ***values of the columns*** we want to combine

```
lang_mother_tidy <- pivot_longer(lang_wide,
  cols = Toronto:Edmonton,
  names_to = "region",
  values_to = "mother_tongue"
)

lang_mother_tidy
```

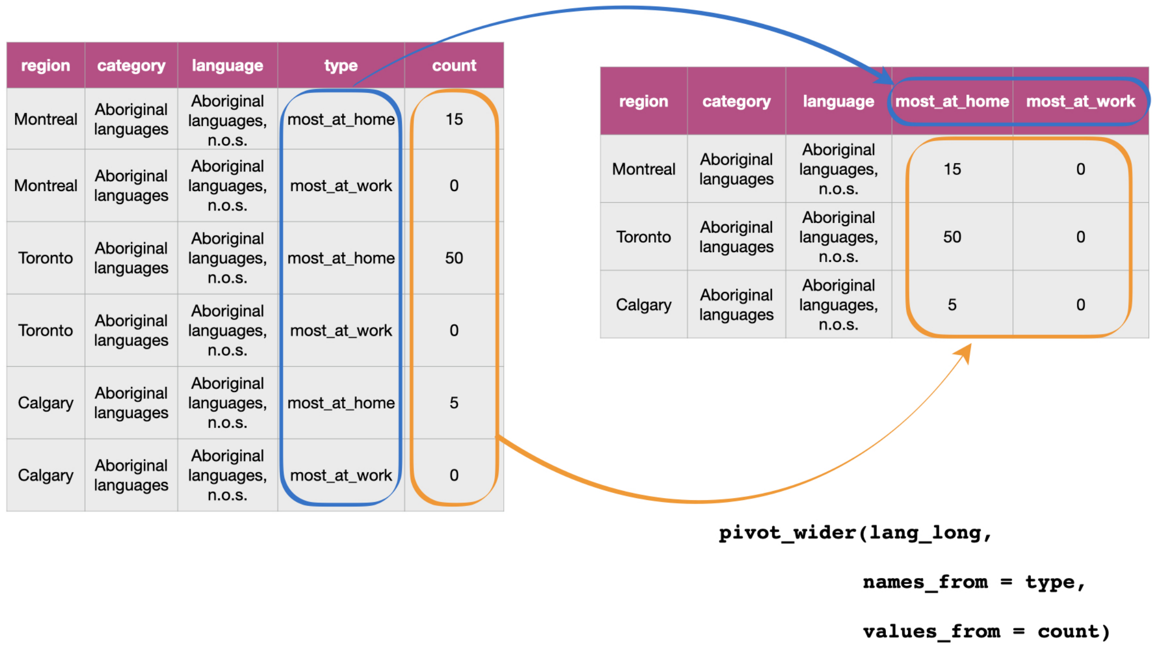
```
## # A tibble: 1,070 x 4
##    category                                  language                region mothe~1
##    <chr>                                  <chr>                  <chr>    <int>
##  1 Aboriginal languages                   Aboriginal languages,~ Toron~      80
##  2 Aboriginal languages                   Aboriginal languages,~ Montr~      30
##  3 Aboriginal languages                   Aboriginal languages,~ Vanco~      70
##  4 Aboriginal languages                   Aboriginal languages,~ Calga~      20
##  5 Aboriginal languages                   Aboriginal languages,~ Edmon~      25
##  6 Non-Official & Non-Aboriginal languages Afrikaans             Toron~     985
##  7 Non-Official & Non-Aboriginal languages Afrikaans             Montr~      90
##  8 Non-Official & Non-Aboriginal languages Afrikaans             Vanco~    1435
##  9 Non-Official & Non-Aboriginal languages Afrikaans             Calga~     960
## 10 Non-Official & Non-Aboriginal languages Afrikaans             Edmon~     575
## # ... with 1,060 more rows, and abbreviated variable name 1: mother_tongue
```

The data above is now tidy because all three criteria for tidy data have now been met:

- All the variables (category, language, region and mother_tongue) are now their own columns in the data frame.
- Each observation, (i.e., each language in a region) is in a single row.
- Each value is a single cell, i.e., its row, column position in the data frame is not shared with another value.

Pivot Wider

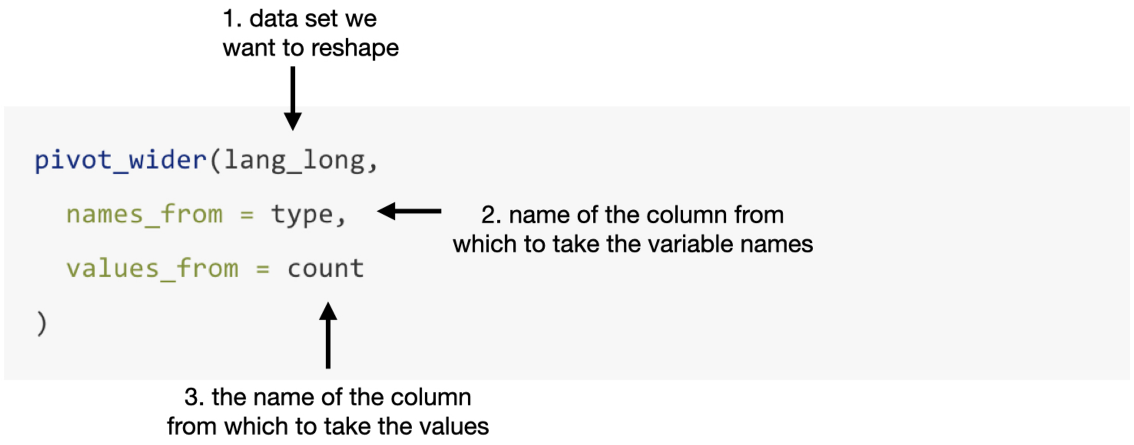
```
lang_long <- read.csv("https://raw.githubusercontent.com/UBC-DSCI/introduction-to-datascience/master/da
```



region	category	language	type	count
Montreal	Aboriginal languages	Aboriginal languages, n.o.s.	most_at_home	15
Montreal	Aboriginal languages	Aboriginal languages, n.o.s.	most_at_work	0
Toronto	Aboriginal languages	Aboriginal languages, n.o.s.	most_at_home	50
Toronto	Aboriginal languages	Aboriginal languages, n.o.s.	most_at_work	0
Calgary	Aboriginal languages	Aboriginal languages, n.o.s.	most_at_home	5
Calgary	Aboriginal languages	Aboriginal languages, n.o.s.	most_at_work	0

region	category	language	most_at_home	most_at_work
Montreal	Aboriginal languages	Aboriginal languages, n.o.s.	15	0
Toronto	Aboriginal languages	Aboriginal languages, n.o.s.	50	0
Calgary	Aboriginal languages	Aboriginal languages, n.o.s.	5	0

```
pivot_wider(lang_long,
            names_from = type,
            values_from = count)
```


1. data set we want to reshape
2. name of the column from which to take the variable names
3. the name of the column from which to take the values

```
pivot_wider(lang_long,
  names_from = type,
  values_from = count
)
```

```
lang_home_tidy <- pivot_wider(lang_long,
  names_from = type,
  values_from = count
)
lang_home_tidy
```

```
## # A tibble: 1,070 x 5
##    region    category                                language    most_~1 most_~2
##    <chr>     <chr>                                   <chr>         <int>   <int>
##  1 Montréal  Aboriginal languages                    Aboriginal~      15       0
##  2 Toronto   Aboriginal languages                    Aboriginal~      50       0
##  3 Calgary   Aboriginal languages                    Aboriginal~       5       0
##  4 Edmonton  Aboriginal languages                    Aboriginal~      10       0
##  5 Vancouver Aboriginal languages                    Aboriginal~      15       0
##  6 Montréal  Non-Official & Non-Aboriginal languages Afrikaans        10       0
##  7 Toronto   Non-Official & Non-Aboriginal languages Afrikaans       265       0
##  8 Calgary   Non-Official & Non-Aboriginal languages Afrikaans       505      15
##  9 Edmonton  Non-Official & Non-Aboriginal languages Afrikaans       300       0
## 10 Vancouver Non-Official & Non-Aboriginal languages Afrikaans       520      10
## # ... with 1,060 more rows, and abbreviated variable names 1: most_at_home,
## #   2: most_at_work
```

Gapminder

```
library(gapminder)
data("gapminder")
```

Let's say we'd like to look at `LifeExp` over time for all the countries in Asia in our dataset.

```
# Create a dataset called asia with the data we need
asia <- gapminder %>%
  filter(continent == "Asia") %>%
  select(country, year, lifeExp)
```

We can create a wide version of our table, where each row is a country and each column a year, with values of `lifeExp` in each cell of the table.

```r
lifeExp_wide <- asia %>%
  # use pivot_wider to go from long to wide format
  pivot_wider(names_from = "year",
              names_prefix = "yr", #it's a good idea to avoid column names that start with a number
              values_from = "lifeExp")
lifeExp_wide
```

```
## # A tibble: 33 x 13
##    country yr1952 yr1957 yr1962 yr1967 yr1972 yr1977 yr1982 yr1987 yr1992 yr1997
##    <fct>    <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
##  1 Afghan~   28.8   30.3   32.0   34.0   36.1   38.4   39.9   40.8   41.7   41.8
##  2 Bahrain   50.9   53.8   56.9   59.9   63.3   65.6   69.1   70.8   72.6   73.9
##  3 Bangla~   37.5   39.3   41.2   43.5   45.3   46.9   50.0   52.8   56.0   59.4
##  4 Cambod~   39.4   41.4   43.4   45.4   40.3   31.2   51.0   53.9   55.8   56.5
##  5 China     44     50.5   44.5   58.4   63.1   64.0   65.5   67.3   68.7   70.4
##  6 Hong K~   61.0   64.8   67.6   70     72     73.6   75.4   76.2   77.6   80
##  7 India     37.4   40.2   43.6   47.2   50.7   54.2   56.6   58.6   60.2   61.8
##  8 Indone~   37.5   39.9   42.5   46.0   49.2   52.7   56.2   60.1   62.7   66.0
##  9 Iran      44.9   47.2   49.3   52.5   55.2   57.7   59.6   63.0   65.7   68.0
## 10 Iraq      45.3   48.4   51.5   54.5   57.0   60.4   62.0   65.0   59.5   58.8
## # ... with 23 more rows, and 2 more variables: yr2002 <dbl>, yr2007 <dbl>
```