Assignment 04: Tidy data and joins

```r
suppressPackageStartupMessages(library(gapminder))
suppressPackageStartupMessages(library(tidyverse))
suppressPackageStartupMessages(library(DT))
suppressPackageStartupMessages(library(scales))
suppressPackageStartupMessages(library(knitr))
```

Exercise 1: Univariate Data Reshaping

Univariate Option 2

Compute some measure of life expectancy (mean? median? min? max?) for all possible combinations of continent and year. Reshape that to have one row per year and one variable for each continent. Or the other way around: one row per continent and one variable per year.

I have chosen to compute a weighted mean (by population) of life expectancy for each continent.

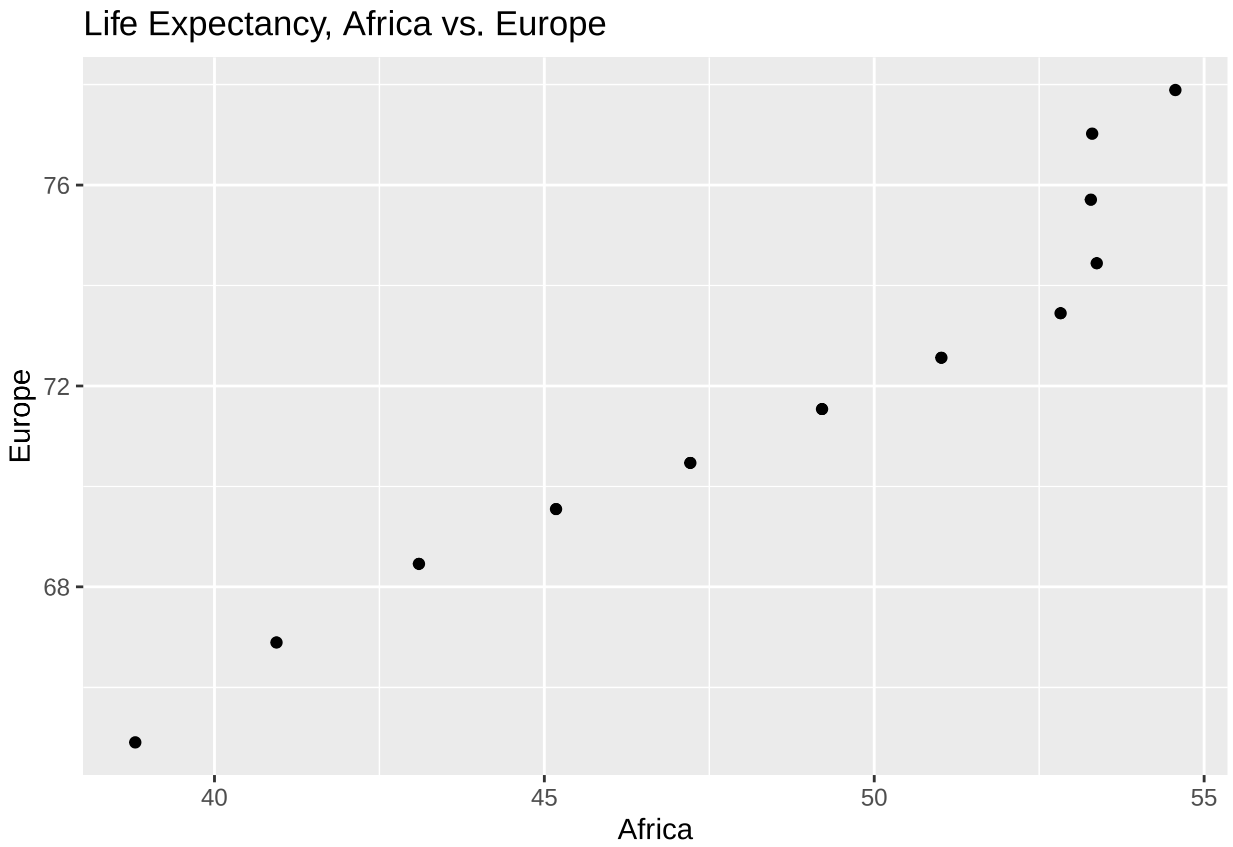
```r
avgContYear <- gapminder %>%
  group_by(continent, year) %>%
  summarize(avgLifeExp=weighted.mean(lifeExp, pop)) %>%
  pivot_wider(id_cols=year,
              names_from=continent,
              values_from=avgLifeExp)
kable(avgContYear)
```

year	Africa	Americas	Asia	Europe	Oceania
1952	38.79973	60.23599	42.94114	64.90540	69.17040
1957	40.94031	62.01806	47.28835	66.89364	70.31693
1962	43.09925	63.43706	46.57369	68.45957	70.98808
1967	45.17721	64.50630	53.88261	69.54963	71.17848
1972	47.21229	65.70490	57.52159	70.46884	71.92273
1977	49.20883	67.60591	59.55648	71.53989	73.25684
1982	51.01744	69.19264	61.57472	72.56247	74.58291
1987	52.82479	70.35814	63.53710	73.44717	75.98107
1992	53.37292	71.72177	65.14874	74.44273	77.35788
1997	53.28327	73.19154	66.77092	75.70849	78.61843
2002	53.30314	74.24736	68.13732	77.02232	80.16006
2007	54.56441	75.35668	69.44386	77.89057	81.06215

Is there a plot that is easier to make with the data in this shape versus the usual form? Try making such a plot!

It is easier to plot the values of variables of the same type against each other, such as in the example shown below, where the life expectancy of Europe is plotted against that of Africa. Both are continents (therefore same type). We can use this type of graph to understand how properties evolved alongside one another.

```r
avgContYear %>%
  ggplot(aes(x=Africa, y=Europe)) +
  geom_point() +
  ggtitle("Life Expectancy, Africa vs. Europe")
```



Re-lengthen the data.

```
avgContYear %>%
  pivot_longer(cols= -year,
               names_to="continent",
               values_to="avgLifeExp")
```

```
## # A tibble: 60 x 3
##     year continent avgLifeExp
##    <int> <chr>          <dbl>
##  1  1952 Africa          38.8
##  2  1952 Americas        60.2
##  3  1952 Asia            42.9
##  4  1952 Europe          64.9
##  5  1952 Oceania         69.2
##  6  1957 Africa          40.9
##  7  1957 Americas        62.0
##  8  1957 Asia            47.3
##  9  1957 Europe          66.9
## 10  1957 Oceania         70.3
## # ... with 50 more rows
```

Exercise 2: Multivariate Data Reshaping

Multivariate Option 1

Make a tibble with one row per year, and columns for life expectancy and GDP per capita (or two other numeric variables) for two or more countries.

Re-lengthen the data.

```
multiwide <- gapminder %>%
  filter(country %in% c("Burundi", "New Zealand", "Switzerland")) %>%
  pivot_wider(id_cols=year,
              names_from=country,
              values_from=c(lifeExp, pop))

kable(multiwide)
```

year	lifeExp_Burundi	lifeExp_New Zealand	lifeExp_Switzerland	pop_Burundi	pop_New Zealand	pop_Switze
1952	39.031	69.390	69.620	2445618	1994794	48
1957	40.533	70.260	70.560	2667518	2229407	51
1962	42.045	71.240	71.320	2961915	2488550	56
1967	43.548	71.520	72.770	3330989	2728150	60
1972	44.057	71.890	73.780	3529983	2929100	64
1977	45.910	72.220	75.390	3834415	3164900	63
1982	47.471	73.840	76.210	4580410	3210650	64
1987	48.211	74.320	77.410	5126023	3317166	66
1992	44.736	76.330	78.030	5809236	3437674	69
1997	45.326	77.550	79.370	6121610	3676187	71
2002	47.360	79.110	80.620	7021078	3908037	73
2007	49.580	80.204	81.701	8390505	4115771	75

```
multiwide %>%
  pivot_longer(cols= -year,
               names_to=c(".value", "country"),
               names_sep="_")
```

```
## # A tibble: 36 x 4
##     year country     lifeExp     pop
##    <int> <chr>         <dbl>   <int>
##  1  1952 Burundi        39.0 2445618
##  2  1952 New Zealand    69.4 1994794
##  3  1952 Switzerland    69.6 4815000
##  4  1957 Burundi        40.5 2667518
##  5  1957 New Zealand    70.3 2229407
##  6  1957 Switzerland    70.6 5126000
##  7  1962 Burundi        42.0 2961915
##  8  1962 New Zealand    71.2 2488550
##  9  1962 Switzerland    71.3 5666000
## 10  1967 Burundi        43.5 3330989
## # ... with 26 more rows
```

Exercise 3: Table Joins (30%)

Read in the made-up wedding guestlist and email addresses using the following lines:

```
suppressMessages(guest <- read_csv("https://raw.githubusercontent.com/STAT545-UBC/Classroom/master/data/
suppressMessages(email <- read_csv("https://raw.githubusercontent.com/STAT545-UBC/Classroom/master/data/
```

3.1

For each guest in the guestlist (guest tibble), add a column for email address, which can be found in the email tibble.

```
email_sep <- email %>%
  separate_rows(guest, sep=", ")

guest %>%
  left_join(email_sep, by=c("name"="guest"))
```

```
## # A tibble: 30 x 8
##    party name  meal_wedding meal_brunch attendance_wedd~ attendance_brun~
##    <dbl> <chr> <chr>        <chr>       <chr>            <chr>
##  1     1 Somm~ PENDING      PENDING     PENDING          PENDING
##  2     1 Phil~ vegetarian   Menu C      CONFIRMED        CONFIRMED
##  3     1 Blan~ chicken      Menu A      CONFIRMED        CONFIRMED
##  4     1 Emaa~ PENDING      PENDING     PENDING          PENDING
##  5     2 Blai~ chicken      Menu C      CONFIRMED        CONFIRMED
##  6     2 Nige~ <NA>         <NA>        CANCELLED        CANCELLED
##  7     3 Sine~ PENDING      PENDING     PENDING          PENDING
##  8     4 Ayra~ vegetarian   Menu B      PENDING          PENDING
##  9     5 Atla~ PENDING      PENDING     PENDING          PENDING
## 10     5 Denz~ fish         Menu B      CONFIRMED        CONFIRMED
## # ... with 20 more rows, and 2 more variables: attendance_golf <chr>,
## #   email <chr>
```

3.2

Who do we have emails for, yet are not on the guestlist?

```
email_sep %>%
  anti_join(guest, by=c("guest"="name")) %>%
  kable()
```

guest	email
Turner Jones	tjjones12@hotmail.ca
Albert Marshall	themarshallfamily1234@gmail.com
Vivian Marshall	themarshallfamily1234@gmail.com

Turner Jones, Albert Marshall, and Vivian Marshall weren't invited to the wedding. :(

3.3

Make a guestlist that includes everyone we have emails for (in addition to those on the original guestlist).

There were three solutions I came up with, based on what data you'd want to keep in the guestlist.

```
# full dataset, guest + emails like in 3.1
guest %>%
  full_join(email_sep, by=c("name"="guest"))
```

```
## # A tibble: 33 x 8
##    party name  meal_wedding meal_brunch attendance_wedd~ attendance_brun~
##    <dbl> <chr> <chr>        <chr>       <chr>            <chr>
##  1     1 Somm~ PENDING      PENDING     PENDING          PENDING
##  2     1 Phil~ vegetarian   Menu C      CONFIRMED        CONFIRMED
##  3     1 Blan~ chicken      Menu A      CONFIRMED        CONFIRMED
##  4     1 Emaa~ PENDING      PENDING     PENDING          PENDING
##  5     2 Blai~ chicken      Menu C      CONFIRMED        CONFIRMED
##  6     2 Nige~ <NA>         <NA>        CANCELLED        CANCELLED
##  7     3 Sine~ PENDING      PENDING     PENDING          PENDING
##  8     4 Ayra~ vegetarian   Menu B      PENDING          PENDING
##  9     5 Atla~ PENDING      PENDING     PENDING          PENDING
## 10     5 Denz~ fish         Menu B      CONFIRMED        CONFIRMED
## # ... with 23 more rows, and 2 more variables: attendance_golf <chr>,
## #   email <chr>
```

```
#same format as guest
guest %>%
  full_join(email_sep %>%
              select(guest),
            by=c("name"="guest"))
```

```
## # A tibble: 33 x 7
##    party name  meal_wedding meal_brunch attendance_wedd~ attendance_brun~
##    <dbl> <chr> <chr>        <chr>       <chr>            <chr>
##  1     1 Somm~ PENDING      PENDING     PENDING          PENDING
##  2     1 Phil~ vegetarian   Menu C      CONFIRMED        CONFIRMED
##  3     1 Blan~ chicken      Menu A      CONFIRMED        CONFIRMED
##  4     1 Emaa~ PENDING      PENDING     PENDING          PENDING
##  5     2 Blai~ chicken      Menu C      CONFIRMED        CONFIRMED
##  6     2 Nige~ <NA>         <NA>        CANCELLED        CANCELLED
##  7     3 Sine~ PENDING      PENDING     PENDING          PENDING
##  8     4 Ayra~ vegetarian   Menu B      PENDING          PENDING
##  9     5 Atla~ PENDING      PENDING     PENDING          PENDING
## 10     5 Denz~ fish         Menu B      CONFIRMED        CONFIRMED
## # ... with 23 more rows, and 1 more variable: attendance_golf <chr>
```

```
# only guest names
guest %>%
  select(guest=name) %>%
  union(select(email_sep, guest))
```

```
## # A tibble: 33 x 1
##    guest
##    <chr>
##  1 Sommer Medrano
##  2 Phillip Medrano
##  3 Blanka Medrano
##  4 Emaan Medrano
##  5 Blair Park
##  6 Nigel Webb
##  7 Sinead English
##  8 Ayra Marks
##  9 Atlanta Connolly
## 10 Denzel Connolly
## # ... with 23 more rows
```