Zeyd Khalil HW8, October 15, 2020

```
library(tidyverse)
```

```
## -- Attaching packages ------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts ---------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Exercise 1 - The following built-in datasets are not tidy. For each one, describe why it is not tidy and write out what the first five entries would look like once it is in a tidy format.

a. relig_income
b. billboard
c. us_rent_income

```
head(relig_income, n = 5)
```

```
## # A tibble: 5 x 11
##   religion `<$10k` `$10-20k` `$20-30k` `$30-40k` `$40-50k` `$50-75k` `$75-100k`
##   <chr>      <dbl>     <dbl>     <dbl>     <dbl>     <dbl>     <dbl>      <dbl>
## 1 Agnostic      27        34        60        81        76       137        122
## 2 Atheist       12        27        37        52        35        70         73
## 3 Buddhist      27        21        30        34        33        58         62
## 4 Catholic     418       617       732       670       638      1116        949
## 5 Don't k~      15        14        15        11        10        35         21
## # ... with 3 more variables: `$100-150k` <dbl>, `>150k` <dbl>, `Don't
## #   know/refused` <dbl>
```

The reason why this data is not tidy is because the column headings are values and not variable names. In order to tidy this data, I would use count() in order to distribute the income intervals in each row, and count the number of cases in those intervals. The result would consist of 3 columns; religion, income interval, and count. That would significantly increase the number of rows, but decrease the number of columns.

```
head(billboard, n = 5)
```

```
## # A tibble: 5 x 79
##   artist track date.entered   wk1   wk2   wk3   wk4   wk5   wk6   wk7   wk8
##   <chr>  <chr> <date>       <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 2 Pac  Baby~ 2000-02-26      87    82    72    77    87    94    99    NA
## 2 2Ge+h~ The ~ 2000-09-02      91    87    92    NA    NA    NA    NA    NA
## 3 3 Doo~ Kryp~ 2000-04-08      81    70    68    67    66    57    54    53
## 4 3 Doo~ Loser 2000-10-21      76    76    72    69    67    65    55    59
## 5 504 B~ Wobb~ 2000-04-15      57    34    25    17    17    31    36    49
## # ... with 68 more variables: wk9 <dbl>, wk10 <dbl>, wk11 <dbl>, wk12 <dbl>,
## #   wk13 <dbl>, wk14 <dbl>, wk15 <dbl>, wk16 <dbl>, wk17 <dbl>, wk18 <dbl>,
## #   wk19 <dbl>, wk20 <dbl>, wk21 <dbl>, wk22 <dbl>, wk23 <dbl>, wk24 <dbl>,
## #   wk25 <dbl>, wk26 <dbl>, wk27 <dbl>, wk28 <dbl>, wk29 <dbl>, wk30 <dbl>,
## #   wk31 <dbl>, wk32 <dbl>, wk33 <dbl>, wk34 <dbl>, wk35 <dbl>, wk36 <dbl>,
## #   wk37 <dbl>, wk38 <dbl>, wk39 <dbl>, wk40 <dbl>, wk41 <dbl>, wk42 <dbl>,
## #   wk43 <dbl>, wk44 <dbl>, wk45 <dbl>, wk46 <dbl>, wk47 <dbl>, wk48 <dbl>,
## #   wk49 <dbl>, wk50 <dbl>, wk51 <dbl>, wk52 <dbl>, wk53 <dbl>, wk54 <dbl>,
## #   wk55 <dbl>, wk56 <dbl>, wk57 <dbl>, wk58 <dbl>, wk59 <dbl>, wk60 <dbl>,
## #   wk61 <dbl>, wk62 <dbl>, wk63 <dbl>, wk64 <dbl>, wk65 <dbl>, wk66 <lgl>,
## #   wk67 <lgl>, wk68 <lgl>, wk69 <lgl>, wk70 <lgl>, wk71 <lgl>, wk72 <lgl>,
## #   wk73 <lgl>, wk74 <lgl>, wk75 <lgl>, wk76 <lgl>
```

The reason why this data is not tidy is the same as in the previous example, because most of the columns in this dataset are numerical values instead of variable names. In order to tidy this data, I would use count() in order to distribute the week number in each row, and count the number of cases in those intervals. The result would consist of 4 columns; artist, track, date entered, week number, and count. That would significantly increase the number of rows, but decrease the number of columns.

```
head(us_rent_income, n = 5)
```

```
## # A tibble: 5 x 5
##   GEOID NAME    variable estimate   moe
##   <chr> <chr>   <chr>       <dbl> <dbl>
## 1 01    Alabama income      24476   136
## 2 01    Alabama rent          747     3
## 3 02    Alaska  income      32940   508
## 4 02    Alaska  rent         1200    13
## 5 04    Arizona income      27517   148
```

The reason why this data is not tidy is because each observational unit is spread across multiple rows. In order to make this data tidy, what I would do is use mutate() in order to create a new column for rent. Once that's done, the first five entries of the data will consist of 5 columns and only one row per state; the columns will be GEOID, NAME, Income, Rent, and moe.

Exercise 2 - Try on your own to do the same thing to table4b.

```
tidy4b <- table4b %>% pivot_longer(cols = c(`1999`, `2000`), names_to = "year", values_to = "cases")

tidy4b
```

```
## # A tibble: 6 x 3
##   country     year       cases
##   <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
```

Exercise 3 - Tidy built-in dataset relig_income

```
tidy_relig_income <- relig_income %>% pivot_longer(cols = c(`<$10k`, `$10-20k`, `$20-30k`, `$30-40k`, `$
tidy_relig_income
```

```
## # A tibble: 180 x 3
##    religion `religion income`  count
##    <chr>    <chr>              <dbl>
##  1 Agnostic <$10k                 27
##  2 Agnostic $10-20k               34
##  3 Agnostic $20-30k               60
##  4 Agnostic $30-40k               81
##  5 Agnostic $40-50k               76
##  6 Agnostic $50-75k              137
##  7 Agnostic $75-100k             122
##  8 Agnostic $100-150k            109
##  9 Agnostic >150k                 84
## 10 Agnostic Don't know/refused    96
## # ... with 170 more rows
```

Exercise 4 -