Assignment_04 Final

Steven Tran

February 28, 2018

```
library(tidyverse)
```

```
## -- Attaching packages ----------------------------------------------------- tidyverse 1.2.1 --
```

```
## v ggplot2 2.2.1     v purrr   0.2.4
## v tibble  1.4.2     v dplyr   0.7.4
## v tidyr   0.8.0     v stringr 1.2.0
## v readr   1.1.1     v forcats 0.2.0
```

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

```
library(dplyr)
```

R for Data Science

10.5 Exercises

5. What does tibble::enframe() do? When might you use it?

enframe() converts vectors or lists to a dataframe. The opposite to this is deframe(). I would use enframe() when I am given a vector or list to analyze.

12.6 Exercises

3. I claimed that iso2 and iso3 were redundant with country. Confirm this claim.

```
head(who)
```

```
## # A tibble: 6 x 60
##   country     iso2  iso3   year new_sp_m014 new_sp_m1524 new_sp_m2534
##   <chr>       <chr> <chr> <int>       <int>        <int>        <int>
## 1 Afghanistan AF    AFG    1980          NA           NA           NA
## 2 Afghanistan AF    AFG    1981          NA           NA           NA
## 3 Afghanistan AF    AFG    1982          NA           NA           NA
## 4 Afghanistan AF    AFG    1983          NA           NA           NA
## 5 Afghanistan AF    AFG    1984          NA           NA           NA
## 6 Afghanistan AF    AFG    1985          NA           NA           NA
## # ... with 53 more variables: new_sp_m3544 <int>, new_sp_m4554 <int>,
## #   new_sp_m5564 <int>, new_sp_m65 <int>, new_sp_f014 <int>,
## #   new_sp_f1524 <int>, new_sp_f2534 <int>, new_sp_f3544 <int>,
## #   new_sp_f4554 <int>, new_sp_f5564 <int>, new_sp_f65 <int>,
## #   new_sn_m014 <int>, new_sn_m1524 <int>, new_sn_m2534 <int>,
## #   new_sn_m3544 <int>, new_sn_m4554 <int>, new_sn_m5564 <int>,
## #   new_sn_m65 <int>, new_sn_f014 <int>, new_sn_f1524 <int>,
```

```
## #   new_sn_f2534 <int>, new_sn_f3544 <int>, new_sn_f4554 <int>,
## #   new_sn_f5564 <int>, new_sn_f65 <int>, new_ep_m014 <int>,
## #   new_ep_m1524 <int>, new_ep_m2534 <int>, new_ep_m3544 <int>,
## #   new_ep_m4554 <int>, new_ep_m5564 <int>, new_ep_m65 <int>,
## #   new_ep_f014 <int>, new_ep_f1524 <int>, new_ep_f2534 <int>,
## #   new_ep_f3544 <int>, new_ep_f4554 <int>, new_ep_f5564 <int>,
## #   new_ep_f65 <int>, newrel_m014 <int>, newrel_m1524 <int>,
## #   newrel_m2534 <int>, newrel_m3544 <int>, newrel_m4554 <int>,
## #   newrel_m5564 <int>, newrel_m65 <int>, newrel_f014 <int>,
## #   newrel_f1524 <int>, newrel_f2534 <int>, newrel_f3544 <int>,
## #   newrel_f4554 <int>, newrel_f5564 <int>, newrel_f65 <int>
```

```
tail(who)
```

```
## # A tibble: 6 x 60
##   country  iso2  iso3   year new_sp_m014 new_sp_m1524 new_sp_m2534
##   <chr>    <chr> <chr> <int>       <int>        <int>        <int>
## 1 Zimbabwe ZW    ZWE    2008         127          614            0
## 2 Zimbabwe ZW    ZWE    2009         125          578           NA
## 3 Zimbabwe ZW    ZWE    2010         150          710         2208
## 4 Zimbabwe ZW    ZWE    2011         152          784         2467
## 5 Zimbabwe ZW    ZWE    2012         120          783         2421
## 6 Zimbabwe ZW    ZWE    2013          NA           NA           NA
## # ... with 53 more variables: new_sp_m3544 <int>, new_sp_m4554 <int>,
## #   new_sp_m5564 <int>, new_sp_m65 <int>, new_sp_f014 <int>,
## #   new_sp_f1524 <int>, new_sp_f2534 <int>, new_sp_f3544 <int>,
## #   new_sp_f4554 <int>, new_sp_f5564 <int>, new_sp_f65 <int>,
## #   new_sn_m014 <int>, new_sn_m1524 <int>, new_sn_m2534 <int>,
## #   new_sn_m3544 <int>, new_sn_m4554 <int>, new_sn_m5564 <int>,
## #   new_sn_m65 <int>, new_sn_f014 <int>, new_sn_f1524 <int>,
## #   new_sn_f2534 <int>, new_sn_f3544 <int>, new_sn_f4554 <int>,
## #   new_sn_f5564 <int>, new_sn_f65 <int>, new_ep_m014 <int>,
## #   new_ep_m1524 <int>, new_ep_m2534 <int>, new_ep_m3544 <int>,
## #   new_ep_m4554 <int>, new_ep_m5564 <int>, new_ep_m65 <int>,
## #   new_ep_f014 <int>, new_ep_f1524 <int>, new_ep_f2534 <int>,
## #   new_ep_f3544 <int>, new_ep_f4554 <int>, new_ep_f5564 <int>,
## #   new_ep_f65 <int>, newrel_m014 <int>, newrel_m1524 <int>,
## #   newrel_m2534 <int>, newrel_m3544 <int>, newrel_m4554 <int>,
## #   newrel_m5564 <int>, newrel_m65 <int>, newrel_f014 <int>,
## #   newrel_f1524 <int>, newrel_f2534 <int>, newrel_f3544 <int>,
## #   newrel_f4554 <int>, newrel_f5564 <int>, newrel_f65 <int>
```

No matter which observation one picks, iso2 and iso3 changes accordingly with country and is redundant.

4. For each country, year, and sex compute the total number of cases of TB. Make an informative visualisation of the data.

```
whoTidy <- who %>%
  gather(code, value, new_sp_m014:newrel_f65, na.rm = TRUE) %>%
  mutate(code = stringr::str_replace(code, "newrel", "new_rel")) %>%
  separate(code, c("new", "var", "sexage")) %>%
  select(-new, -iso2, -iso3) %>%
  separate(sexage, c("sex", "age"), sep = 1) %>%
  group_by(country, year, sex) %>%
```

```
  summarize(Number =n())
whoTidy
```

```
## # A tibble: 6,921 x 4
## # Groups:   country, year [?]
##    country      year sex   Number
##    <chr>       <int> <chr>  <int>
##  1 Afghanistan  1997 f          7
##  2 Afghanistan  1997 m          7
##  3 Afghanistan  1998 f          7
##  4 Afghanistan  1998 m          7
##  5 Afghanistan  1999 f          7
##  6 Afghanistan  1999 m          7
##  7 Afghanistan  2000 f          7
##  8 Afghanistan  2000 m          7
##  9 Afghanistan  2001 f          7
## 10 Afghanistan  2001 m          7
## # ... with 6,911 more rows
```