Exercise 12.6.1

Xiru Lyu

2/20/2018

```
# load in required package
library(tidyverse)
```

1. In this case study I set `na.rm = TRUE` just to make it easier to check that we had the correct values. Is this reasonable? Think about how missing values are represented in this dataset. Are there implicit missing values? What's the difference between an `NA` and zero?

Removal of `NA` values in the dataset isn't reasonable. `NA` values mean that there is missing value on new case of disease in one country in one particular year. If we want to study the trend in our data, or to report that some of the data is missing, we want to include `NA` values so that we are sure to have data for each year in each country. There might also be implicit missing values, for example, there might be cases that data of some years in some countries are not recorded in the datatable. Or it's possible that for all countries in all years there is no new case coming up for one gender in one particular age group. All these can be considered as implicit missing values. The difference between an `NA` and zero is that `NA` tells us that the data is missing, meaning that the actual value (if to be found) can be either zero or non-zero. However, zero tells us that there is no observation falling within one category.

2. What happens if you neglect the `mutate()` step? (`mutate(key = stringr::str_replace(key, "newrel", "new_rel"))`)

If `mutate()` step is neglected, then the `separate()` step cannot be performed accurately as 'newrel' would be recognized as the value in column 'new', the number indicating sex and age group would be filled in the column 'type' and there would be `NA` generated for the column 'sexage'. Moreover, for data in this specific row, value `NA` cannot be further separated to sex and age. This would lead to wrong data.

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

```
# import the dataset

who <- 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"))

# check to see if `iso2` and `iso3` are redundant
who %>% select(country, iso2,iso3) %>% distinct() %>% group_by(country) %>% filter(n() > 1)
```

```
## # A tibble: 0 x 3
## # Groups:   country [0]
## # ... with 3 variables: country <chr>, iso2 <chr>, iso3 <chr>
```

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

```
who <- who %>% select(-iso2,-iso3,-new) %>% separate(sexage,c('sex','age'),sep=1)

who %>% group_by(country,year,sex) %>% summarize(cases=sum(value)) %>% ggplot(aes(x=year,y=cases,group=
```