Exercises 12.6.1

Tiffany Cheng

February 20, 2018

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?

I think to quickly check if the values were correct, removing the NAs is reasonable. However, I think that the NAs in this dataset represent both explicit and implicit missing values, so removing them all would be a mistake. `NA` is used when there is no data for the variable recorded while a zero could be a actual data observation.

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

If you neglect the `mutate()` step, then separating the `key` column into `new`, `type`, and `sexage` would require an extra step since `newrel` is not separated by a "_". This step makes the data more consistent.

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

```
## # A tibble: 7,240 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
##  7 Afghanistan    AF   AFG  1986          NA           NA           NA
##  8 Afghanistan    AF   AFG  1987          NA           NA           NA
##  9 Afghanistan    AF   AFG  1988          NA           NA           NA
## 10 Afghanistan    AF   AFG  1989          NA           NA           NA
## # ... with 7,230 more rows, and 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>
```

```
who1 <- who %>%
  gather(new_sp_m014:newrel_f65, key = "key", value = "cases", na.rm = TRUE)
%>%
  mutate(key = stringr::str_replace(key, "newrel", "new_rel")) %>%
  separate(key, c("new", "type", "sexage"), sep = "_") %>%
  count(iso2,iso3)

head(who1)
```

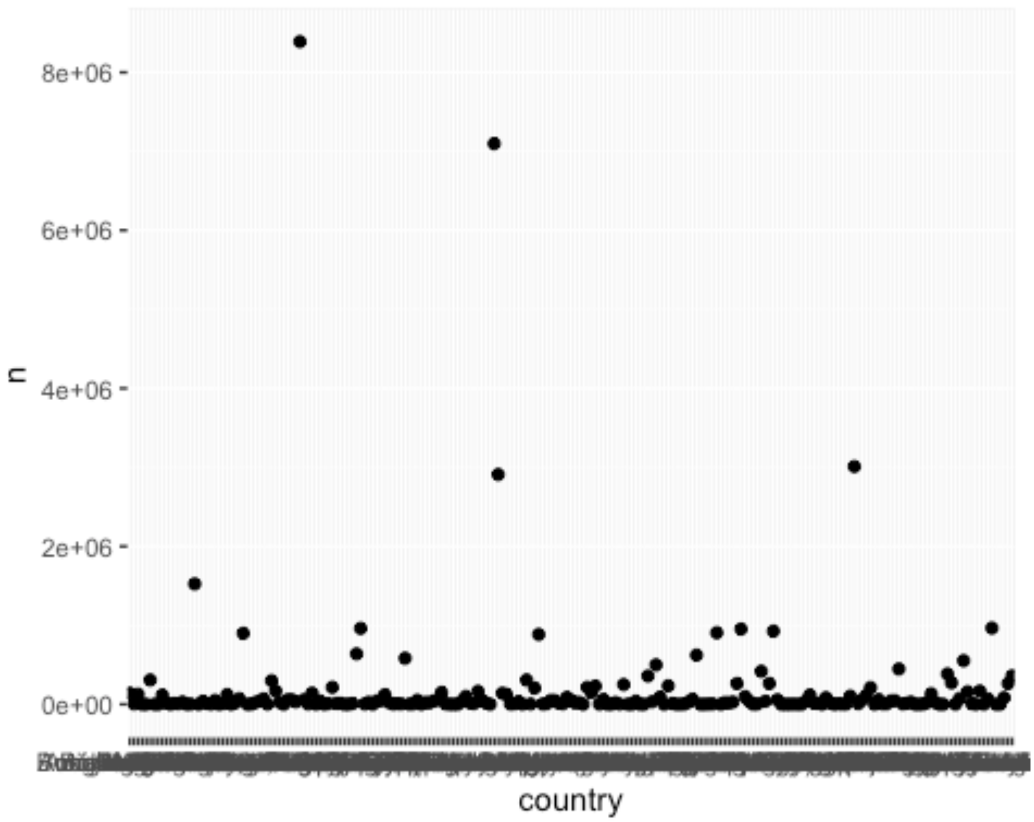
```
## # A tibble: 6 x 3
##    iso2  iso3     n
##   <chr> <chr> <int>
## 1    AD   AND   387
## 2    AE   ARE   378
## 3    AF   AFG   244
## 4    AG   ATG   346
## 5    AI   AIA   155
## 6    AL   ALB   448
```

`iso2` seems to be a two-letter abbreviation for the country name and `iso3` seems to be a three-letter abbreviation for the country name. Therefore, if we keep the `country` column, these two column are redundant because they carry the same information.

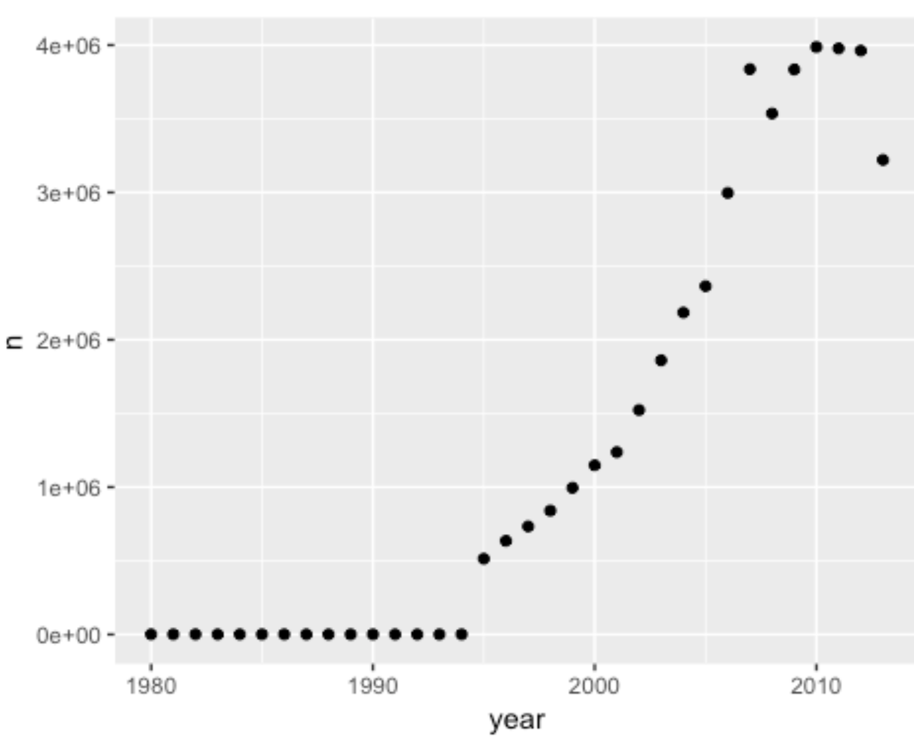
4. For each country, year, and sex compute the total number of cases of TB. Make an informative visualisation of the data.

```
# Tidying data.
who2 <- who %>%
  gather(new_sp_m014:newrel_f65, key = "key", value = "cases", na.rm = TRUE)
%>%
  mutate(key = stringr::str_replace(key, "newrel", "new_rel")) %>%
  separate(key, c("new", "type", "sexage"), sep = "_") %>%
  select(-new, -iso2, -iso3) %>%
  separate(sexage, c("sex", "age"), sep = 1)

# Calculating number of cases (n) by country, year, and sex.
country <- who2 %>%
```

```
  count(country, wt=cases)
head(country)

## # A tibble: 6 x 2
##          country      n
##            <chr>  <int>
## 1    Afghanistan 140225
## 2        Albania   5335
## 3        Algeria 128119
## 4 American Samoa     41
## 5        Andorra    103
## 6         Angola 308365

year <- who2 %>%
  count(year, wt=cases)
head(year)

## # A tibble: 6 x 2
##    year     n
##   <int> <int>
## 1  1980   959
## 2  1981   805
## 3  1982   824
## 4  1983   786
## 5  1984   814
## 6  1985   799

sex <- who2 %>%
  count(sex, wt=cases)
head(sex)

## # A tibble: 2 x 2
##     sex        n
##   <chr>    <int>
## 1     f 15907024
## 2     m 27490494

# Creating informative visualizations of the data.
ggplot(data=country) +
  geom_point(mapping=aes(x=country,y=n))
```




```
ggplot(data=year) +
  geom_point(mapping=aes(x=year,y=n))
```



```
ggplot(data=sex) +
  geom_point(mapping=aes(x=sex,y=n))
```