

Assignment4

LingzhouAo

4/14/2018

```
library(foreign)
library(stringr)
library(plyr)
library(reshape2)
suppressMessages(library(tidyverse))
```

12.6.1

```
who1 <- who %>%
  gather(new_sp_m014:newrel_f65, key = "key", value = "cases", na.rm = TRUE)

who2 <- who1 %>%
  mutate(key = stringr::str_replace(key, "newrel", "new_rel"))

who3 <- who2 %>%
  separate(key, c("new", "type", "sexage"), sep = "_")

who3 %>%
  count(new)

## # A tibble: 1 x 2
##   new      n
##   <chr> <int>
## 1 new   76046

who4 <- who3 %>%
  select(-new, -iso2, -iso3)

who5 <- who4 %>%
  separate(sexage, c("sex", "age"), sep = 1)
```

Problem3

The iso2 and iso3 are 2&3 letter ISO country codes, and because the full name of country can represent a specific country, we can regard them as redundant information and drop them.

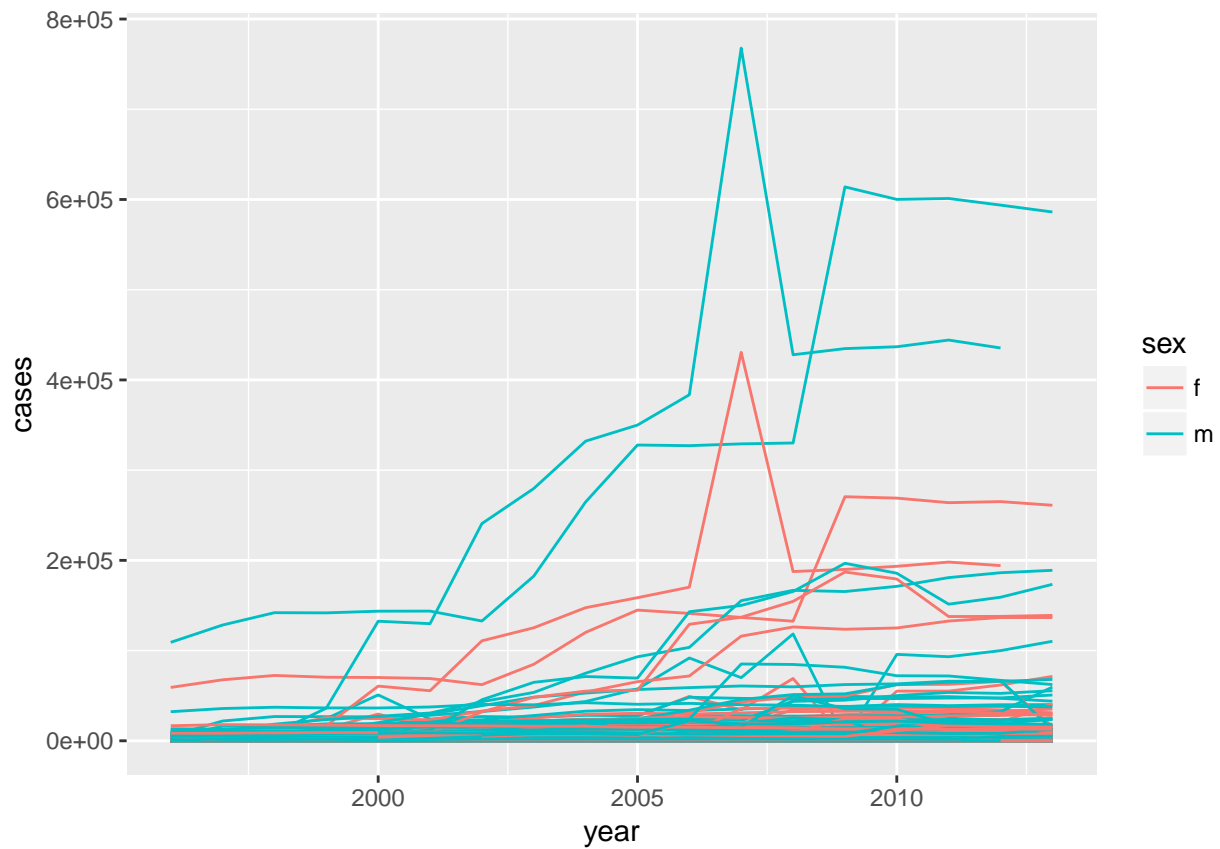
```
select(who3, country, iso2, iso3) %>%
  distinct() %>%
  group_by(country) %>%
  filter(n() > 1)
```

```
## # A tibble: 0 x 3
```

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

Problem4

```
who5 %>%
  group_by(country, year, sex) %>%
  filter(year > 1995) %>%
  summarise(cases = sum(cases)) %>%
  unite(country_sex, country, sex, remove = FALSE) %>%
  ggplot(aes(x = year, y = cases, group = country_sex, colour = sex)) +
  geom_line()
```



10.5

Problem5

`enframe()` converts named atomic vectors or lists to two-column data frames. For unnamed vectors, the natural sequence is used as name column.

```
?tibble::enframe
enframe(c(a = 5, b = 7))

## # A tibble: 2 x 2
##   name  value
##   <chr> <dbl>
## 1 a      5.
## 2 b      7.
```

Tabel4->Table6

```
pew <- read.spss("pew.sav")

## re-encoding from CP1252

## Warning in read.spss("pew.sav"): Undeclared level(s) 2, 3, 4, 9 added in
## variable: density3

## Warning in read.spss("pew.sav"): Duplicated levels in factor denom:
## Electronic ministries

## Warning in read.spss("pew.sav"): Undeclared level(s) 1, 2, 3, 4, 5, 6, 7,
## 8, 9, 10, 11, 12, 14, 16, 23, 33 added in variable: children

## Warning in read.spss("pew.sav"): Undeclared level(s) 18, 19, 20, 21, 22,
## 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41,
## 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60,
## 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79,
## 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 added in
## variable: age

pew <- as.data.frame(pew)

religion <- pew[c("q16", "reltrad", "income")]
religion$reltrad <- as.character(religion$reltrad)
religion$reltrad <- str_replace(religion$reltrad, " Churches", "")
religion$reltrad <- str_replace(religion$reltrad, " Protestant", " Prot")
religion$reltrad[religion$q16 == " Atheist (do not believe in God) "] <- "Atheist"
religion$reltrad[religion$q16 == " Agnostic (not sure if there is a God) "] <- "Agnostic"
religion$reltrad <- str_trim(religion$reltrad)
religion$reltrad <- str_replace_all(religion$reltrad, " \\(.*?\\)", "")

religion$income <- c("Less than $10,000" = "<$10k",
                    "10 to under $20,000" = "$10-20k",
```

```

    "20 to under $30,000" = "$20-30k",
    "30 to under $40,000" = "$30-40k",
    "40 to under $50,000" = "$40-50k",
    "50 to under $75,000" = "$50-75k",
    "75 to under $100,000" = "$75-100k",
    "100 to under $150,000" = "$100-150k",
    "$150,000 or more" = ">150k",
    "Don't know/Refused (VOL)" = "Don't know/refused")[religion$income]

religion$income <- factor(religion$income, levels = c("<$10k", "$10-20k", "$20-30k", "$30-40k", "$40-50k",
    "$75-100k", "$100-150k", ">150k", "Don't know/refused"))

counts <- plyr::count(religion, c("reltrad", "income"))
names(counts)[1] <- "religion"
table4 <- dcast(counts, religion ~ income)

## Using freq as value column: use value.var to override.

table6 <- gather(table4, key = "income", value = "freq", -religion)
table6 <- arrange(table6, religion)

```

```
t7 <- read_csv("billboard.csv")

## Parsed with column specification:
## cols(
##   .default = col_integer(),
##   artist.inverted = col_character(),
##   track = col_character(),
##   time = col_time(format = ""),
##   genre = col_character(),
##   date.entered = col_date(format = ""),
##   date.peaked = col_date(format = ""),
##   x66th.week = col_character(),
##   x67th.week = col_character(),
##   x68th.week = col_character(),
##   x69th.week = col_character(),
##   x70th.week = col_character(),
##   x71st.week = col_character(),
##   x72nd.week = col_character(),
##   x73rd.week = col_character(),
##   x74th.week = col_character(),
##   x75th.week = col_character(),
##   x76th.week = col_character()
## )

## See spec(...) for full column specifications.

t7 <- t7[, c("year", "artist.inverted", "track", "time", "date.entered", "x1st.week", "x2nd.week", "x3rd.week", "x4th.week", "x5th.week", "x6th.week", "x7th.week", "x8th.week", "x9th.week", "x10th.week", "x11th.week", "x12th.week", "x13th.week", "x14th.week", "x15th.week", "x16th.week", "x17th.week", "x18th.week", "x19th.week", "x20th.week", "x21st.week", "x22nd.week", "x23rd.week", "x24th.week", "x25th.week", "x26th.week", "x27th.week", "x28th.week", "x29th.week", "x30th.week", "x31st.week", "x32nd.week", "x33rd.week", "x34th.week", "x35th.week", "x36th.week", "x37th.week", "x38th.week", "x39th.week", "x40th.week", "x41st.week", "x42nd.week", "x43rd.week", "x44th.week", "x45th.week", "x46th.week", "x47th.week", "x48th.week", "x49th.week", "x50th.week", "x51st.week", "x52nd.week", "x53rd.week", "x54th.week", "x55th.week", "x56th.week", "x57th.week", "x58th.week", "x59th.week", "x60th.week", "x61st.week", "x62nd.week", "x63rd.week", "x64th.week", "x65th.week", "x66th.week", "x67th.week", "x68th.week", "x69th.week", "x70th.week", "x71st.week", "x72nd.week", "x73rd.week", "x74th.week", "x75th.week", "x76th.week")]
names(t7)[2] <- "artist"

t7$artist <- iconv(t7$artist)
```

```

t7$track <- str_replace(t7$track, "\\(..*?\\)", "")
names(t7)[-1:5] <- str_c("wk", 1:76)
t7 <- arrange(t7, year, artist, track)

long_name <- nchar(t7$track) > 20
t7$track[long_name] <- paste0(substr(t7$track[long_name], 0, 20), "...")

table7 <- t7[c(1:3, 6:10), ]

t8 <- table7%>%gather(key = 'week', value = 'rank', -year, -artist, -time, -track, -date.entered, na.rm = TRUE)

t8$week <- as.integer(str_replace_all(t8$week, "[^0-9]+", ""))
names(t8)[5] <- "date"
t8 <- t8[order(t8$artist),]

```