wednesday class notes

Tim Riffe

7/7/2021

summary

On the first day we looked at the crude birth rate and the crude death rate. The question was posed as to what constitutes high and low, or what constitutes a comparatively / relatively high and low rate.

For example, take a 10-year-wide age group. The mortality rate increases a lot from age 70 to 79. If the population weights happen to be heavier in the bottom of the interval, it will pull a 10-year rate down, and vice versa. But even 10-year age groups are better than none!!!

So in general, we can't fully account for all of the ways that population heterogeneity can perturb a crude rate. And here I'm referring even to rates in age groups as *crude* because even then there are likely other ways to stratify that would reveal meaningful population-level mortality differences. So what story do we need to tell ourselves to be able to think that a give mortality rate actually reflects underlying risk conditions? That's super tricky to wrap your mind around because 1) everyone dies at a different age, even in perfectly homogenous subgroups (stochasticity) 2) composition can be important, but we should be able to assume that it's somewhat constant within the specific strata, or steady in some sense (stationary would be the technical term). I have to abandon this paragraph as an incomplete thought.

Data

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.2     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

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

```
library(readr)
# will copy this link into the google doc too
DAT <- read_csv("https://raw.githubusercontent.com/timriffe/BSSD2021Module2/master/03_wednesday/wednesda
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Country = col_character(),
##   Year = col_double(),
##   Sex = col_character(),
##   Age = col_double(),
##   Exposure = col_double(),
##   M = col_double()
## )
```

```
head(DAT)
```

```
## # A tibble: 6 x 6
##   Country  Year Sex      Age Exposure        M
##   <chr>   <dbl> <chr>  <dbl>    <dbl>    <dbl>
## 1 Taiwan   1970 female     0  180000  0.0154
## 2 Taiwan   1970 female     1  186891. 0.00446
## 3 Taiwan   1970 female     2  183319. 0.00306
## 4 Taiwan   1970 female     3  187248. 0.00193
## 5 Taiwan   1970 female     4  195642. 0.00119
## 6 Taiwan   1970 female     5  199660. 0.000833
```

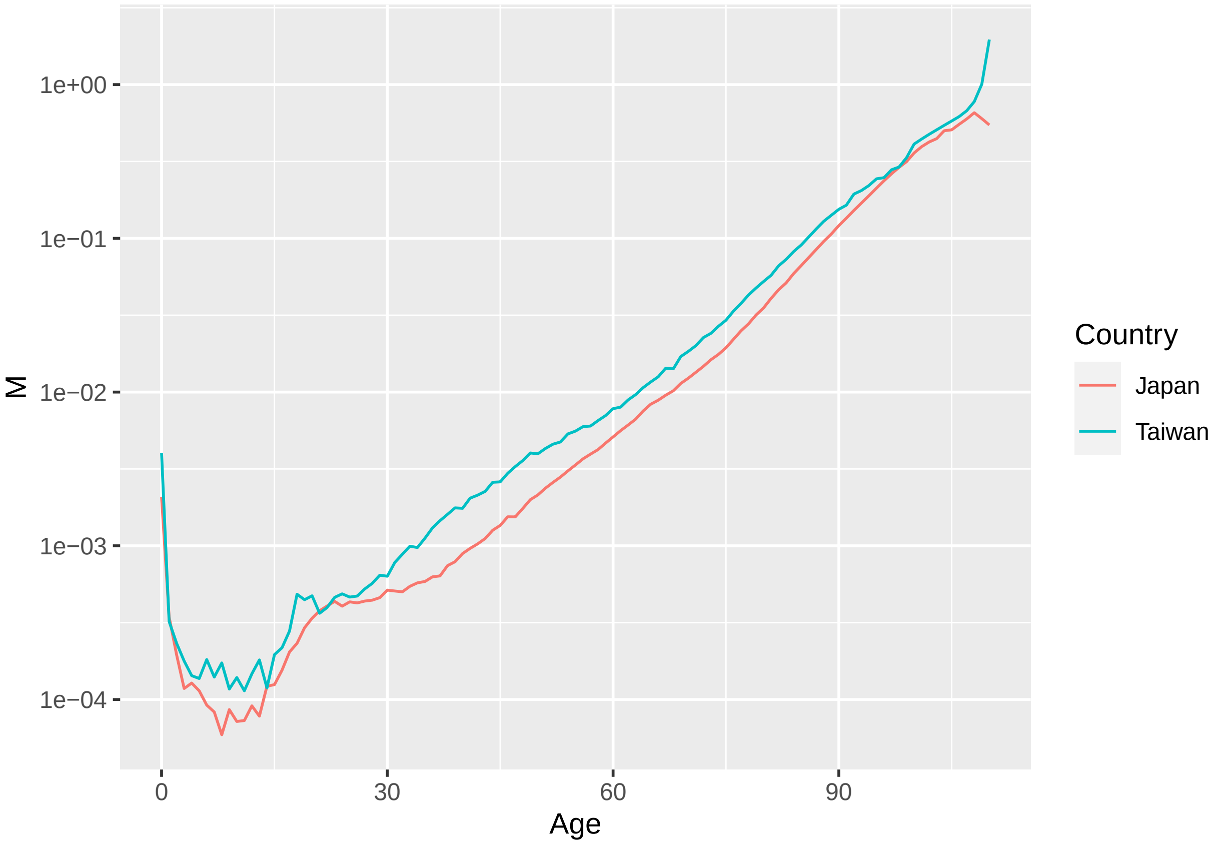
the crude rates we would like to explain

What do mortality rates look like?

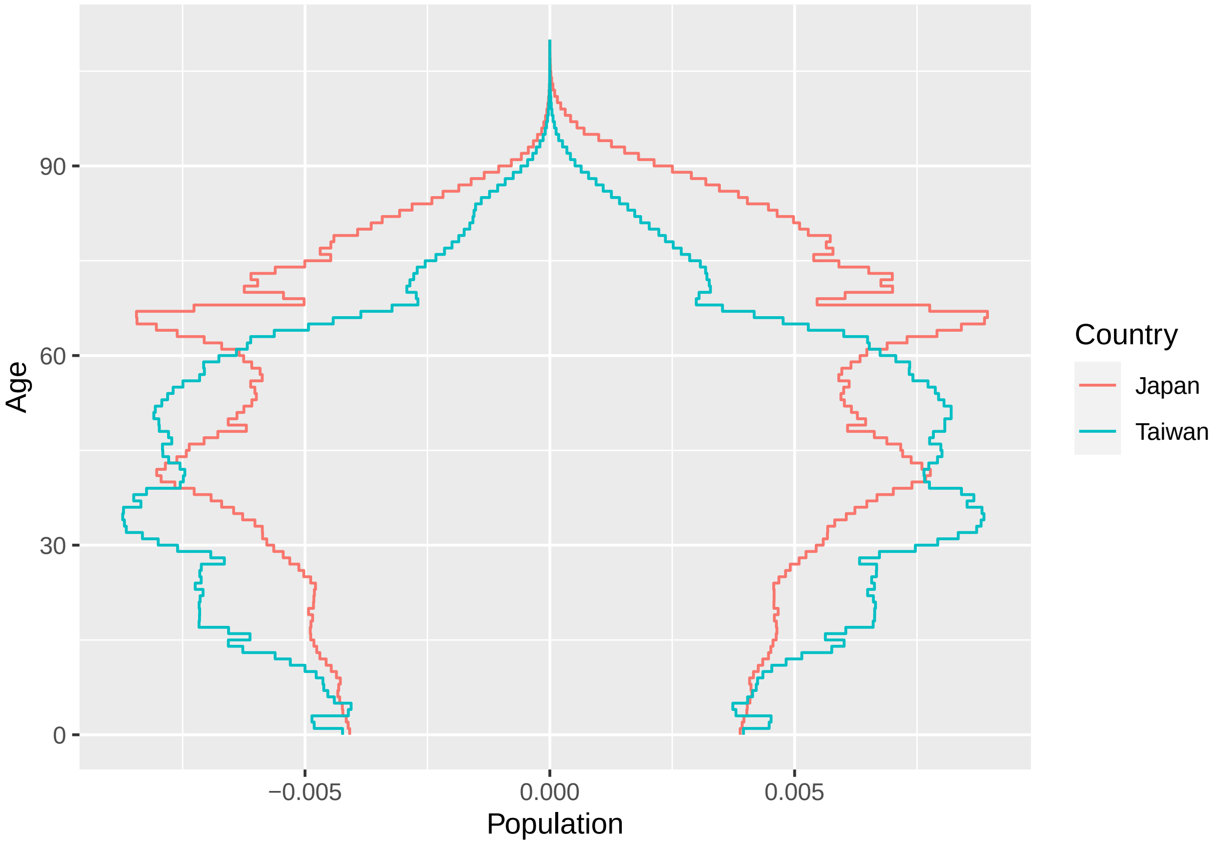
```
DAT %>%
  filter(Sex == "total",
         Year == 2014) %>%
  group_by(Country) %>%
  summarize(CDR = 1000 * sum(Exposure * M) / sum(Exposure))
```

```
## # A tibble: 2 x 2
##   Country   CDR
##   <chr>   <dbl>
## 1 Japan   10.1
## 2 Taiwan   6.98
```

```
DAT %>%
  filter(Sex == "total",
         Year == 2014) %>%
  ggplot(aes(x = Age, y = M, color = Country)) +
  geom_line() +
  scale_y_log10()
```



```
DAT %>%
  filter(Year == 2014,
         Sex != "total") %>%
  group_by(Country) %>%
  mutate(Structure = Exposure / sum(Exposure),
         Population = ifelse(Sex == "male", -Structure, Structure)) %>%
  ungroup() %>%
  ggplot(aes(x = Age,
             y = Population,
             color = Country,
             group = interaction(Country, Sex))) +
  geom_step() +
  coord_flip()
```



We know from looking at these pictures that the reason CDR is higher in Japan is due entirely to population structure because 1) Taiwan rates seem to be higher in every age, and 2) we can see that a relatively higher fraction of the Japanese population is in older age groups.

direct standardization

For this we ask what would the CDR be if structure were the same for both populations? To do this, we often that the mean structure of both populations as the standard, but sometimes there are reasons or traditions that suggest using other standards.

```
# step 1, get structure for each country, and take it's average
DAT2 <-
  DAT %>%
  filter(Year == 2014,
         Sex == "total") %>%
  group_by(Country) %>%
  mutate(Structure = Exposure / sum(Exposure))

ST <-
  DAT2 %>%
  group_by(Age) %>%
  summarize(Standard = mean(Structure))

# step 2, join the standard to the data
DAT2 %>%
  left_join(ST, by = "Age") %>%
  group_by(Country) %>%
  summarize(CDR = 1000 * sum(M * Structure),
            ASDR = 1000 * sum(M * Standard))
```

```
## # A tibble: 2 x 3
##   Country   CDR  ASDR
##   <chr>   <dbl> <dbl>
## 1 Japan   10.1   7.45
## 2 Taiwan   6.98 10.6
```

indirect standardization

```
DAT2 %>%
  group_by(Age) %>%
  summarize(M_standard = mean(M)) %>%
  ungroup() %>%
  right_join(DAT2, by = "Age") %>%
  group_by(Country) %>%
  summarize(CDR = 1000 * sum(M * Structure),
            ASDR_i = 1000 * sum(M_standard * Structure))
```

```
## # A tibble: 2 x 3
##   Country   CDR ASDR_i
##   <chr>   <dbl>  <dbl>
## 1 Japan   10.1   12.2
## 2 Taiwan   6.98   5.87
```

Kitagawa decomposition

In words: the difference in two crude rates is equal to the sum of the difference in rates times the average structure plus the sum of the difference in structure times the average rates.

```
DAT %>%
  filter(Year == 2014,
         Sex == "total") %>%
  group_by(Country) %>%
  mutate(Sx = Exposure / sum(Exposure)) %>%
  ungroup() %>%
  select(-Exposure) %>%
  pivot_wider(names_from = Country, values_from = c(M, Sx)) %>%
  mutate(avg_M = (M_Japan + M_Taiwan) / 2,
         avg_Sx = (Sx_Japan + Sx_Taiwan) / 2,
         RE = (M_Japan - M_Taiwan) * avg_Sx,
         CE = (Sx_Japan - Sx_Taiwan) * avg_M) %>%
  summarize(RE = 1000 * sum(RE),
            CE = 1000 * sum(CE),
            CDR_Japan = 1000 * sum(M_Japan * Sx_Japan),
            CDR_Taiwan = 1000 * sum(M_Taiwan * Sx_Taiwan)) %>%
  mutate(CDR_diff = CDR_Japan - CDR_Taiwan,
         Diff_check = RE+CE)
```

```
## # A tibble: 1 x 6
##       RE    CE CDR_Japan CDR_Taiwan CDR_diff Diff_check
##    <dbl> <dbl>     <dbl>      <dbl>    <dbl>      <dbl>
## 1 -3.13  6.28      10.1       6.98     3.16       3.16
```

Final Kitagawa note: it's more general than at first glance, but I'm not yet sure how general it can get.

more decomposition.

There appeared to be a consensus to deliver generalized decomposition, possibly due to my excitement. Sorry Arriaga I really like your method too.

To apply general decomposition you need to be able to calculate your result of interest based on a set of parameters. For life expectancy that clearly means mortality rates. But for other things it could be different.

More specifically, you need to be able to stick your calculations inside a nice little function.

For that reason, we should review how to write a function

```
my_hacky_e0_function <- function(mx){
  H  <- cumsum(mx)
  lx <- c(1,exp(-H))
  e0 <- sum(lx) - .5
  e0
}

DAT2 %>%
  filter(Country == "Taiwan") %>%
  pull(M) %>%
  my_hacky_e0_function()
```

```
## [1] 79.54816
```

Rules for writing a decomposable function (in the `R` sense). The function needs to depend on a vector of parameters. However that gets soted out inside the function is YOUR OWN BUSINESS.

```
my_CrudeRate <- function(params){

  # first we need to sort out which parameter is which
  n <- length(params)
  # we stacked M on top of Sx, so reshape to a 2 column matrix
  dim(params) <- c(n / 2, 2)

  # sum(params[,1] * params[,2])

  M  <- params[, 1]
  Sx <- params[, 2]
  1000 * sum(M * Sx)
}
```

Here we have another function, but it needs two pieces of information: rates and structure. So, instead of making two arguments `M` and `Sx` I've stacked them in a vector `c(M, Sx)`, so the very first thing we need to do inside our function is sort out which piece is which. The function should return the summary measure: a length 1 vector, a scalar. Not a whole vector or something weird like that.

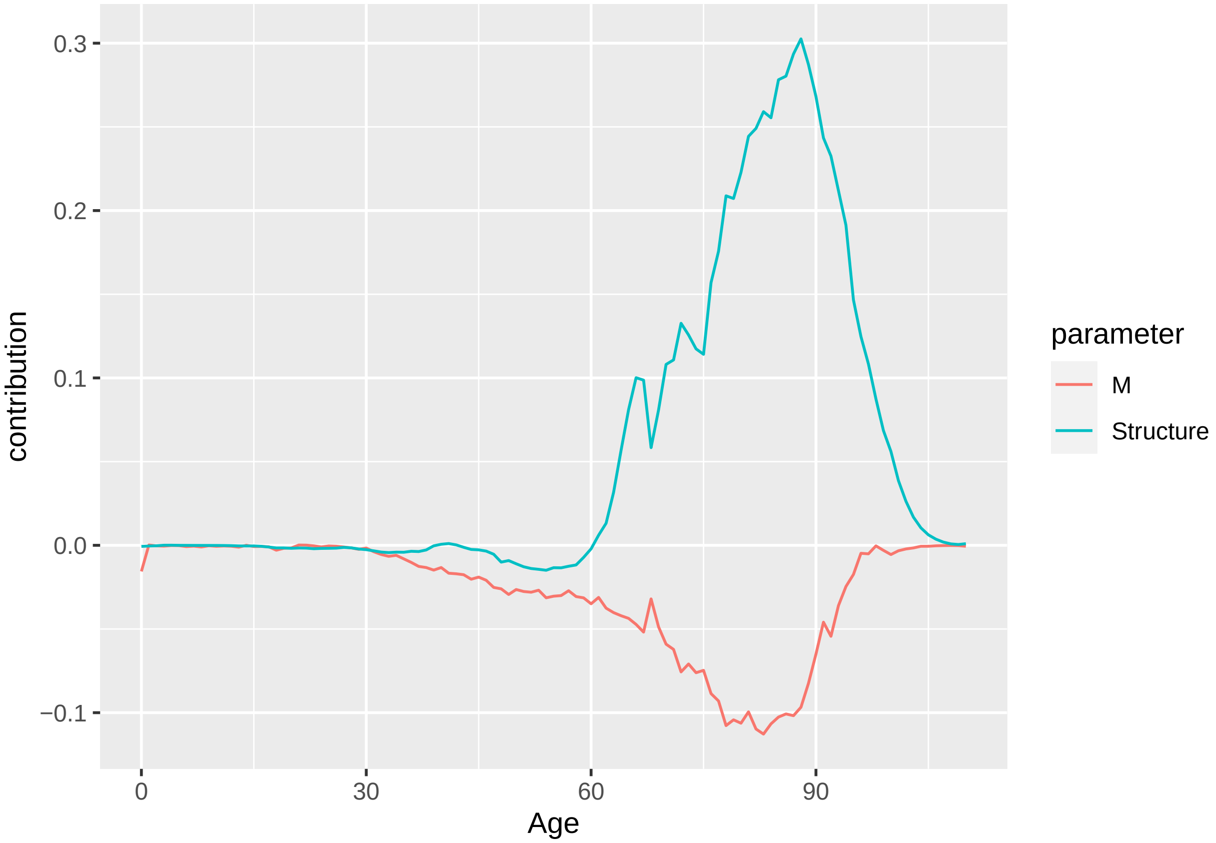
```
# install.packages("DemoDecomp")
```

```
library(DemoDecomp)
DEC <-
DAT2 %>%
  select(-Exposure) %>%
  pivot_longer(M:Structure,
               names_to = "parameter",
               values_to = "pars") %>%
  arrange(Country, parameter, Age) %>%
  pivot_wider(names_from = Country, values_from = pars) %>%
  mutate(contribution = horiuchi(my_CrudeRate, Taiwan, Japan, N = 10))

DEC$contribution %>% sum()
```

```
## [1] 3.157305
```

```
DEC %>%
  ggplot(aes(x = Age, y = contribution, color = parameter)) +
  geom_line()
```



```
DEC %>%
  group_by(parameter) %>%
  summarize(contribution = sum(contribution))
```

```
## # A tibble: 2 x 2
##   parameter contribution
##   <chr>            <dbl>
## 1 M                -3.13
## 2 Structure         6.28
```

Other decomposition options in this package include `ltre()`, `stepwise()`