Session 1 notes

Tim Riffe

2025-07-08

Intro to this document

When I write here, this is just free text, as if we were in Word or whatever.

Ctrl + Alt + i

Time out for installing latex to be able to build pdfs!!

```
# this is R, careful now!
# install.packages("tinytex")
library(tinytex)
#install_tinytex()
```

What is a population?

Just remember to always be clear about population universes. This is especially evident when numerators and denominators come from different sources.

Demography

Contrast between the ends of the accounting mandate and scientific inquiry, a difference between the ends but not the means; same data reliance, but perhaps different levels of flexibility with selecting sources.

Rates vs probabilities

Probability:

$$probability = \frac{events}{\text{number entering exposure}}$$

Rates:

$$rate = \frac{events}{\text{annualized exposure}}$$

annualized exposure can be a rather extreme idea. Say you want monthly age-specific fertility rates, and you know the annual exposure, but not the monthly one? Then you need to discount the annual exposure (make it a lot smaller) based on the number of days in the month relative to a year.

$$rate_{feb} = \frac{events_{feb}}{exposure_{year} \cdot \frac{28}{365}}$$

Time intervals: With rates, we need to be careful about what we mean by annualized exposure. With demographic probabilities, we need to be careful about time horizons and how they might map to magnitudes of probabilities. Longer time horizons = more probable demographic event. Conversely, short time horizons (less than a year) shrink probabilities.

Load data

Ctrl + Enter to execute a line (Cmnd +Enter on mac)

```
library(tidyverse)

D <- read_csv("https://raw.githubusercontent.com/timriffe/BSSD2025Module2/refs/heads/master/data/ES_D20
B <- read_csv("https://raw.githubusercontent.com/timriffe/BSSD2025Module2/refs/heads/master/data/ES_B20
P <- read_csv("https://raw.githubusercontent.com/timriffe/BSSD2025Module2/refs/heads/master/data/ES_P20
```

Calculate exposures somehow

The first method here is the intuitive solution, because you get the two years side-by-side, but what if we had a longer time series? Then what would we do?

```
P |>
  select(-open_interval) |>
  pivot_longer(female:total, names_to = "sex", values_to = "p") |>
  pivot_wider(names_from = year, values_from = p) |>
  mutate(exposure = (`2014` + `2015`) / 2)
```

```
## # A tibble: 333 x 5
##      age sex    `2014` `2015` exposure
##    <dbl> <chr>   <dbl>  <dbl>    <dbl>
##  1     0 female 205612 205537  205574.
##  2     0 male   217832 219373  218602.
##  3     0 total  423444 424910  424177
##  4     1 female 218704 206301  212502.
##  5     1 male   232780 218499  225640.
##  6     1 total  451484 424800  438142
##  7     2 female 228326 218590  223458
##  8     2 male   242703 232496  237600.
##  9     2 total  471029 451086  461058.
## 10     3 female 232229 227786  230008.
## # i 323 more rows
```

Second, more general, method. Create side-by-side series using lags. Now, if you had a longer time series of years, the code would look the same to do the same calculation.

```
E <-
  P |>
  select(-open_interval) |>
  pivot_longer(female:total, names_to = "sex", values_to = "p1") |>
  arrange(sex,age,year) |>
  group_by(sex, age) |>
  mutate(p2 = lead(p1) ) |>
```

```
  filter(!is.na(p2)) |>
  mutate(exposure = (p1 + p2) / 2) |>
  ungroup()
```

Calculate crude death rates

For the sake of not cluttering the environment, we string two `left_join()` statements in a pipeline, where each object is modified in place as it is specified to the join arguments. Notice that `age = join_by(age)` was a helpful message that went to the console. I specfied it afterwards. It's more rigourous to use the `by` argument, so always a good idea to update the code.

```
ES2014 <-
  left_join(
  # cut down exposures to total only (left)
  E |>
  filter(sex == "total") |>
  select(age, exposure),
  # select needed columns from deaths (right)
  D |>  select(age, deaths = total),
  by = join_by(age)) |>
  # the incoming data object from above is the "left"
  left_join(
    # select only needed columns from births
    B |>  select(age, births = total),
    by = join_by(age)
  ) |>
  mutate(births = if_else(is.na(births),0, births))
```

Crude rates are a 1-liner:

```
ES2014 |>
  summarize(CDR = 1000 * sum(deaths) / sum(exposure),
            CBR = 1000 * sum(births) / sum(exposure))
```

```
## # A tibble: 1 x 2
##     CDR   CBR
##   <dbl> <dbl>
## 1  8.48  9.17
```

$$\text{weighted mean} = \bar{x} = \frac{\sum x_i \cdot w_i}{\sum w_i}$$

If w adds up to 1, then:

$$\bar{x} = \sum x_i \cdot w_i$$

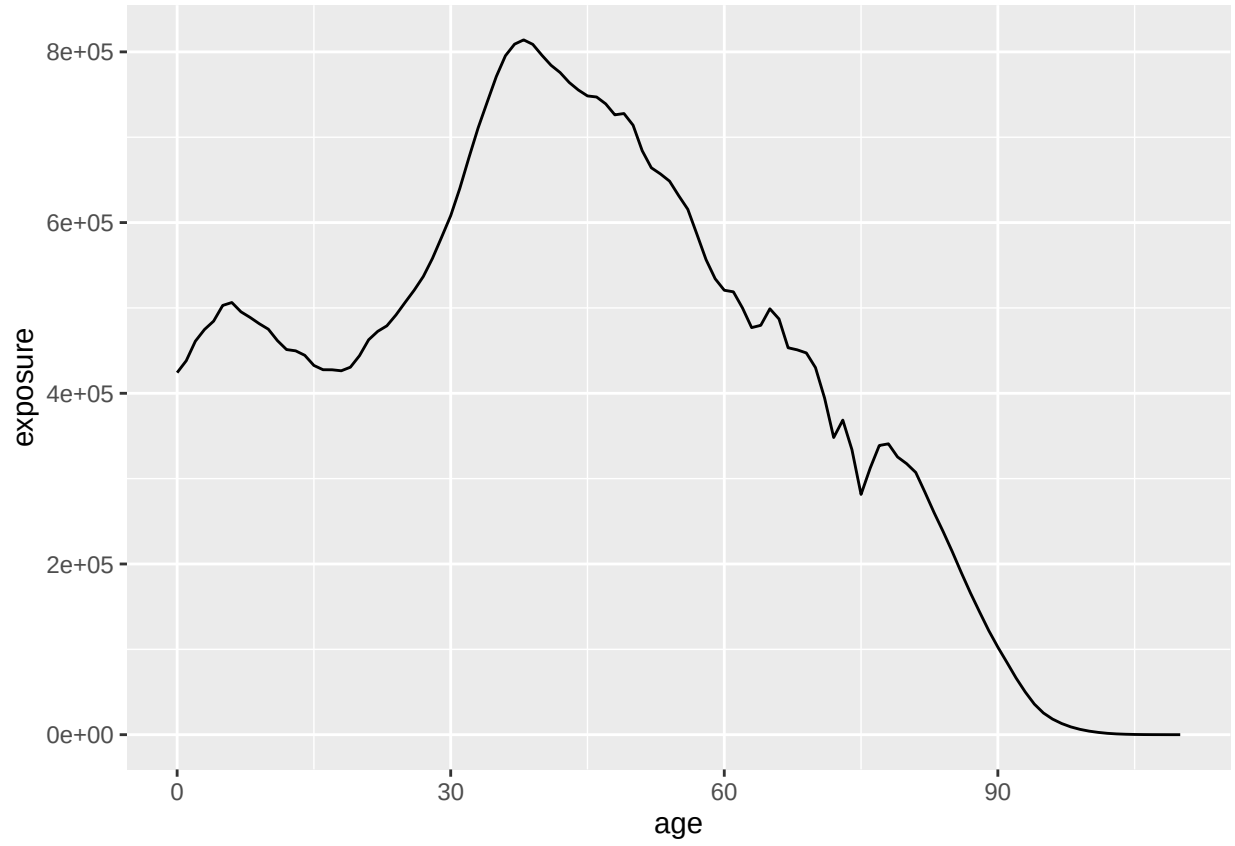
Challenge: Demonstrate to me that the crude rates are also weighted means, where population structure is the weight (w) and age-specific rates are the thing being weighted (x).

Steps: in `mutate()` calculate w and rates in `summarize()` calculate the weight mean of the rates using the formula above. Verify that it's the same as our original result for CBR, CDR.

```r
ES2014 |>
  mutate(mx = deaths / exposure,
         w = exposure / sum(exposure)) |>
  #pull(w) |> sum()
  summarize(CDR = 1000 * sum(mx * w))
```

```
## # A tibble: 1 x 1
##     CDR
##   <dbl>
## 1  8.48
```

```r
ES2014 |>
  ggplot(aes(x = age, y = exposure)) +
  geom_line()
```

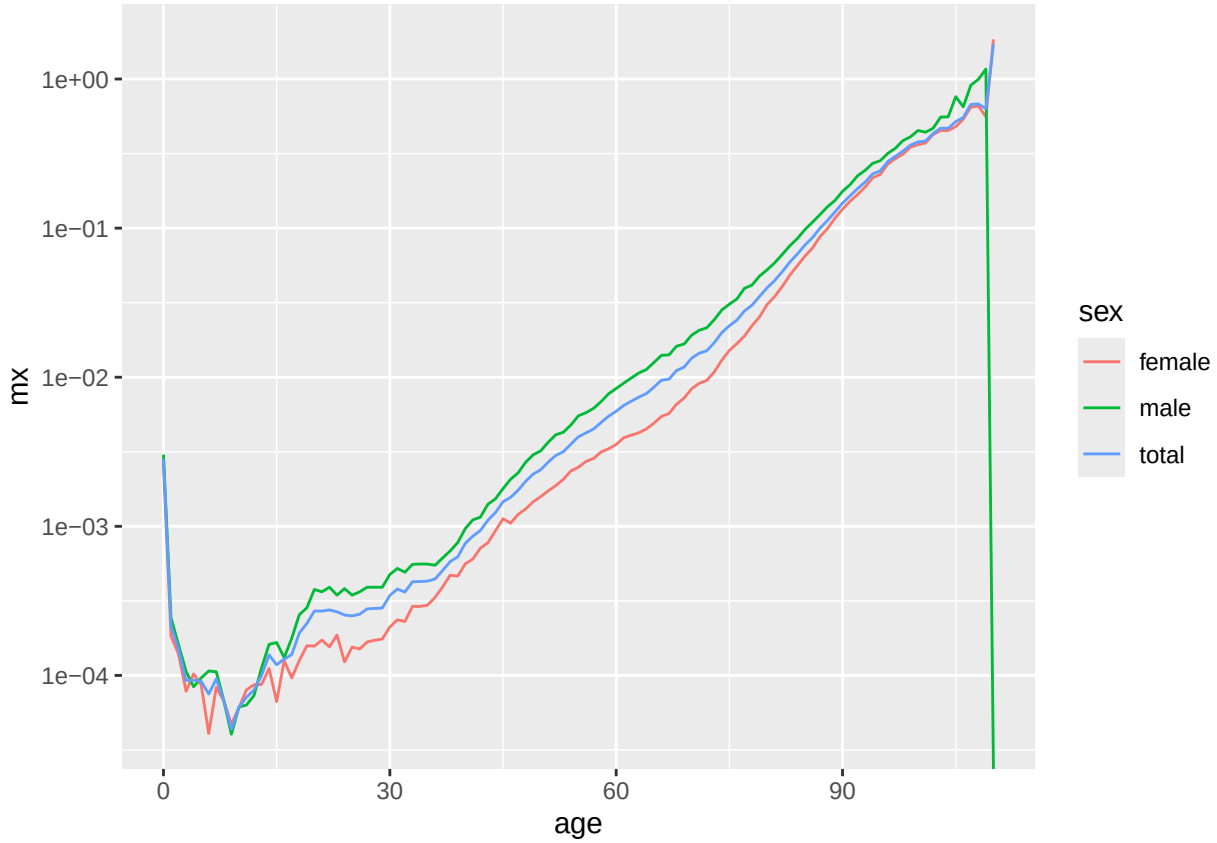


shape of mortality over age

```r
D2 <-
  D |>
  select(-open_interval) |>
  pivot_longer(female:total, names_to = "sex", values_to = "deaths")
```

```
E |>
  select(-p1,-p2) |>
  left_join(D2, by = join_by(year, sex, age)) |>
  mutate(mx = deaths / exposure) |>
  ggplot(mapping = aes(x = age,
                       y = mx,
                       color = sex)) +
  geom_line() +
  scale_y_log10()
```

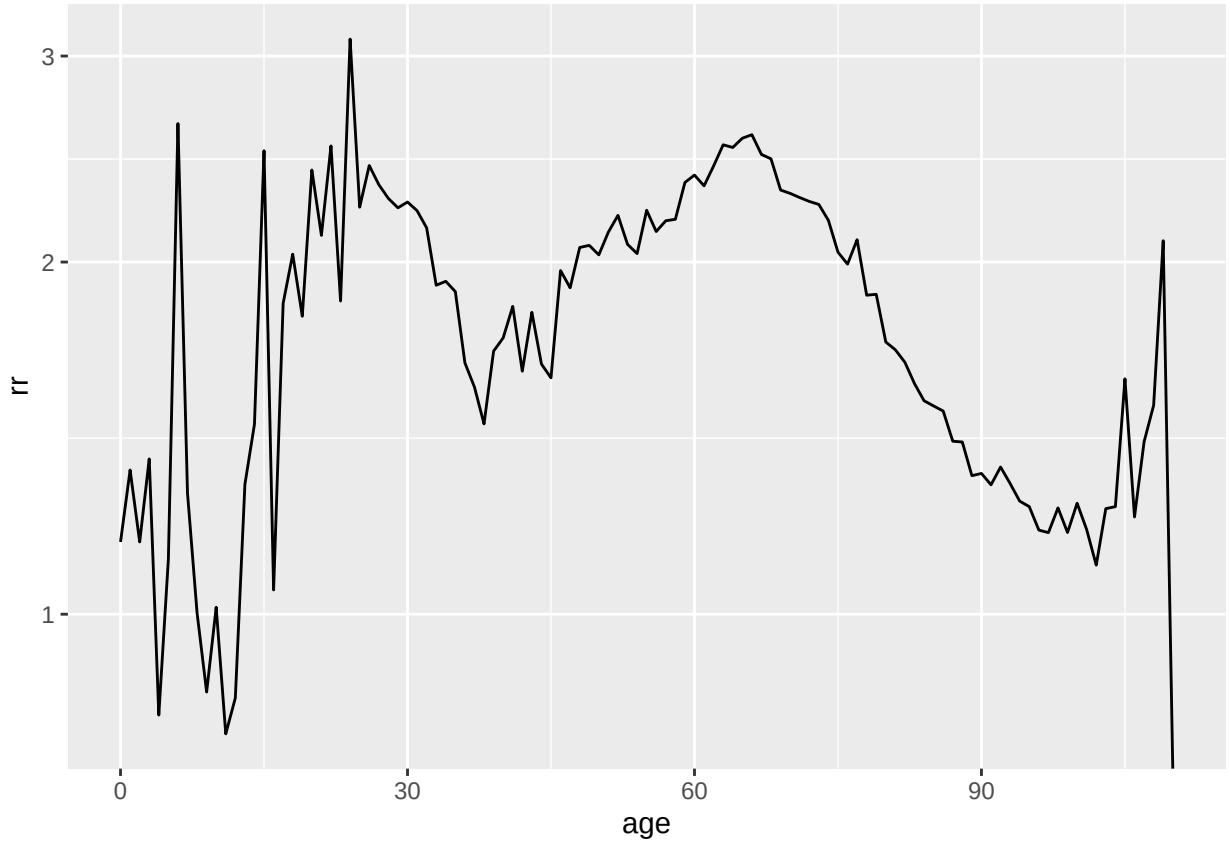
```
## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
```



Challenge: plot the sex ratio of mortality (log scale, male / female)

```
E |>
  select(-p1,-p2) |>
  left_join(D2, by = join_by(year, sex, age)) |>
  mutate(mx = deaths / exposure) |>
  select(-deaths, -exposure) |>
  pivot_wider(names_from = sex, values_from = mx) |>
  mutate(rr = male / female) |>
  ggplot(aes(x = age, y = rr)) +
  geom_line() +
  scale_y_log10()
```

```
## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
```



Exercises in session

1. Merge `B` and `Ex`, matching ages, but taking care that births are matched to female exposures. This is how I joined the data
2. Calculate the General Fertility Rate, GFR, which is:

$$GFR = 1000 \cdot \frac{\sum_{15}^{49} B_x}{\sum_{15}^{49} E_x}$$

3. Make a plot of age-specific fertility rates
4. Calculate TFR:

$$TFR = \sum F_x$$

   where:

$$F_x = \frac{B_x}{E_x}$$

5. Calculate the mean age at childbearing, once using births as the weight, and once using fertility rates as the weight.

$$MAB = \frac{\sum F_x \cdot (x + .5)}{\sum F_x}$$

6. Make a plot comparing fertility rates and births as weights to understand the difference in MAB versions.