

DEMG609: Problem Set 4

Nick Graetz

October 11, 2017

A.(10)

```
# Load data for problem set
raw_data <- fread("C:/Users/ngraetz/Documents/repos/demg609/hw3_data_clean.csv")
raw_data[, x := as.numeric(x)]
raw_data[, nNx := gsub(',', '', nNx)]
raw_data[, nNx := as.numeric(nNx)]
raw_data[, nDx := gsub(',', '', nDx)]
raw_data[, nDx := as.numeric(nDx)]

##### calculate_life_table #####
### Define function that calculates complete life table given arguments:
### data = data.table with numeric columns:
### x = beginning of age interval
### nNx = total people reaching age interval
### nDx = total deaths in age interval
### radix = numbers of survivors at age 0 (or births per year in a
### stationary population); defaults to 100000
#####
calculate_life_table <- function(data, radix = 100000) {
# 1. Calculate nmx
data[, nmx := nDx / nNx]

# 2. Calculate nax given Coale and Demeny equations for ages <5 and n/2 for others.
# For 1a0
data[x == 0 & nmx >= .107, nax := 0.35]
data[x == 0 & nmx < .107, nax := 0.053 + (2.8 * nmx)]
# For 4a1
m0 <- data[x == 0, nmx]
data[x == 1 & shift(nmx, 1, type='lag') >= .107, nax := 1.361]
data[x == 1 & shift(nmx, 1, type='lag') < .107, nax := 1.522 - (1.518 * m0)]
# All other age groups
data[x > 1, nax := 5 / 2]
data[x == 85, nax := 1/nmx]

# 3. Calculate nqx
data[x == 0, n := 1]
data[x == 1, n := 4]
data[x > 1, n := 5]
data[, nqx := (n * nmx) / (1 + ((n - nax) * nmx))]
data[x == 85, nqx := 1]

# 4. Calculate npx
data[, npx := 1 - nqx]

# 5. Calculate lx
data[, lx := radix]
for(r in 2:length(data[, lx])) {
previous_lx <- data[r-1, lx]
previous_npx <- data[r-1, npx]
```

```

    data[r, lx := previous_lx * previous_npx]
  }

# 6. Calculate ndx
for(r in 1:length(data[, lx])) {
  lx_n <- data[r+1, lx]
  data[r, ndx := lx - lx_n]
  if(r == length(data[, lx])) data[r, ndx := lx]
}

# 7. Calculate nLx
for(r in 1:length(data[, lx])) {
  lx_n <- data[r+1, lx]
  data[r, nLx := (n * lx_n) + (nax * ndx)]
  if(r == length(data[, lx])) data[r, nLx := lx / nm]
}

# 8. Calculate Tx
for(r in 1:length(data[, lx])) {
  data[r:length(data[, lx]), Tx := sum(nLx)]
}

# 9. Calculate ex
data[, ex := Tx / lx]

# Return complete life table
return(data)
}

# Calculate standard life table with data for problem set and default radix
lt <- calculate_life_table(data = raw_data)

# A.(10)
# CBR = CDR
# 1 / lt[1, ex]

# Death rate above 60
# 1 / lt[x == 60, ex]

# Mean age at death
# lt[1, ex]

# Given 56,059 births per year in this population, how many people turn 65 each year?
stationary_lt <- calculate_life_table(data = copy(raw_data),
                                     radix = 56059)

# stationary_lt[x == 65, lx]

```

Answer:

Because this is a stationary population the $CDR = CBR = \frac{1}{e_0}$ (0.01291) and the crude growth rate is 0. The death rate above age 60 is 0.04769. The mean age at death is 77.47. Assuming 56,059 births, 49272.9 reach their 65 birthday each year.

B.(1)

```

# Set up mu and births (radix in stationary populations) for each race.

```

```

mu_a <- .08
l0_a <- 5000
mu_b <- .10
l0_b <- 10000

# Life expectancies.
e0_a <- 1/mu_a
e0_b <- 1/mu_b

```

Answer:

Given this is a stationary population, $e_0^o = \frac{1}{\mu}$. Life expectancy for A: 12.5
 Life expectancy for B: 10

B.(2)

```

##### calculate_lt #####
### Define function that calculates complete life table for stationary population given arguments:
### data = data.table with numeric columns:
### x = beginning of age interval
### mu_var = name of variable defining force of mortality
### radix_var = name of lx variable
#####
calculate_lt <- function(data, mu_var, radix_var) {

  # 1. Calculate nmx
  data[, nmx := get(mu_var)]

  # All other age groups
  #data[, nax := 5 / 2]

  # 3. Calculate nqx
  data[, n := 5]
  #data[, nqx := (n * nmx) / (1 + ((n - nax) * nmx))]
  #data[x == 100, nqx := 1]

  # 4. Calculate npx
  #data[, npx := 1 - nqx]
  data[, npx := exp(1)^(-n*nmx)]
  data[, nqx := 1 - npx]
  data[x == 100, nqx := 1]

  # 5. Calculate lx
  data[, lx := get(radix_var)]
  for(r in 2:length(data[, lx])) {
    previous_lx <- data[r-1, lx]
    previous_npx <- data[r-1, npx]
    data[r, lx := previous_lx * previous_npx]
  }

  # 6. Calculate ndx
  for(r in 1:length(data[, lx])) {
    lx_n <- data[r+1, lx]
    data[r, ndx := lx - lx_n]
    if(r == length(data[, lx])) data[r, ndx := lx]
  }
}

```

```

# 7. Calculate nLx
for(r in 1:length(data[, lx])) {
  lx_n <- data[r+1, lx]
  # data[r, nLx := (n * lx_n) + (nax * ndx)]
  # if(r == length(data[, lx])) data[r, nLx := lx / nm]
  data[r, nLx := (lx - lx_n) / nm]
  if(r == length(data[, lx])) data[r, nLx := lx / nm]
}

# 8. Calculate Tx
for(r in 1:length(data[, lx])) {
  data[r:length(data[, lx])-1, Tx := sum(nLx)]
}

# 9. Calculate ex
data[, ex := Tx / lx]

# Don't worry about upper age group
data <- data[x != 100, ]

return(data)
}

lt <- data.table(x = seq(0, 100, 5),
  mu_a = rep(mu_a, 21),
  mu_b = rep(mu_b, 21))

# Construct life table for each race using force of mortality as nm.
lt_a <- calculate_lt(data = copy(lt), mu_var = 'mu_a', radix_var = 'l0_a')
lt_b <- calculate_lt(data = copy(lt), mu_var = 'mu_b', radix_var = 'l0_b')

# Make table to compare people reaching x birthday in each race.
birthdays <- data.table(x = seq(0, 95, 5),
  Race_A = lt_a[, lx],
  Race_B = lt_b[, lx])

```

Answer:

Table 1: Individuals reaching each birthday by race

x	Race_A	Race_B
0	5000.00	10000.00
5	3351.60	6065.31
10	2246.64	3678.79
15	1505.97	2231.30
20	1009.48	1353.35
25	676.68	820.85
30	453.59	497.87
35	304.05	301.97
40	203.81	183.16
45	136.62	111.09
50	91.58	67.38
55	61.39	40.87

x	Race_A	Race_B
60	41.15	24.79
65	27.58	15.03
70	18.49	9.12
75	12.39	5.53
80	8.31	3.35
85	5.57	2.03
90	3.73	1.23
95	2.50	0.75

B.(3)

```
lt_a[, asdr := ndx / nLx]
lt_a[, table := 'Race A']
lt_b[, asdr := ndx / nLx]
lt_b[, table := 'Race B']
combined_lt <- data.table(x = seq(0, 95, 5),
                          nNx = lt_a[, nLx] + lt_b[, nLx],
                          nDx = lt_a[, ndx] + lt_b[, ndx])
combined_lt[, nLx := nNx]
combined_lt[, asdr := nDx / nNx]
combined_lt[, table := 'Race A + B']
combined_lt <- rbind(combined_lt, lt_a, lt_b, fill = TRUE)
```

Answer:

Table 2: Age distribution and deaths by race

x	Race A population	Race A deaths	Race B population	Race B deaths
0	20605.00	1648.40	39346.93	3934.69
5	13811.94	1104.96	23865.12	2386.51
10	9258.42	740.67	14474.93	1447.49
15	6206.11	496.49	8779.49	877.95
20	4160.08	332.81	5325.03	532.50
25	2788.58	223.09	3229.79	322.98
30	1869.24	149.54	1958.97	195.90
35	1252.99	100.24	1188.17	118.82
40	839.91	67.19	720.66	72.07
45	563.01	45.04	437.10	43.71
50	377.39	30.19	265.12	26.51
55	252.97	20.24	160.80	16.08
60	169.57	13.57	97.53	9.75
65	113.67	9.09	59.16	5.92
70	76.19	6.10	35.88	3.59
75	51.07	4.09	21.76	2.18
80	34.24	2.74	13.20	1.32
85	22.95	1.84	8.01	0.80

x	Race A population	Race A deaths	Race B population	Race B deaths
90	15.38	1.23	4.86	0.49
95	10.31	0.82	2.95	0.29

B.(4)

Answer:

Table 3: Age distribution and deaths, races combined

x	Population	Deaths	ASDR
0	59951.93	5583.09	0.09313
5	37677.06	3491.47	0.09267
10	23733.35	2188.17	0.09220
15	14985.59	1374.44	0.09172
20	9485.11	865.31	0.09123
25	6018.38	546.07	0.09073
30	3828.21	345.44	0.09023
35	2441.17	219.06	0.08973
40	1560.57	139.26	0.08924
45	1000.11	88.75	0.08874
50	642.51	56.70	0.08825
55	413.78	36.32	0.08777
60	267.11	23.32	0.08730
65	172.82	15.01	0.08685
70	112.07	9.68	0.08640
75	72.84	6.26	0.08598
80	47.44	4.06	0.08557
85	30.96	2.64	0.08517
90	20.24	1.72	0.08480
95	13.26	1.12	0.08444

B.(5)

Answer:

$$ASDR_{AB} = \frac{(ASDR_A * nN_{x_A}) + (ASDR_B * nN_{x_B})}{nN_{x_A} + nN_{x_B}}$$

$$\frac{nD_{x_A} + nD_{x_B}}{nN_{x_A} + nN_{x_B}} = \frac{(\frac{nD_{x_A}}{nN_{x_A}} * nN_{x_A}) + (\frac{nD_{x_B}}{nN_{x_B}} * nN_{x_B})}{nN_{x_A} + nN_{x_B}}$$

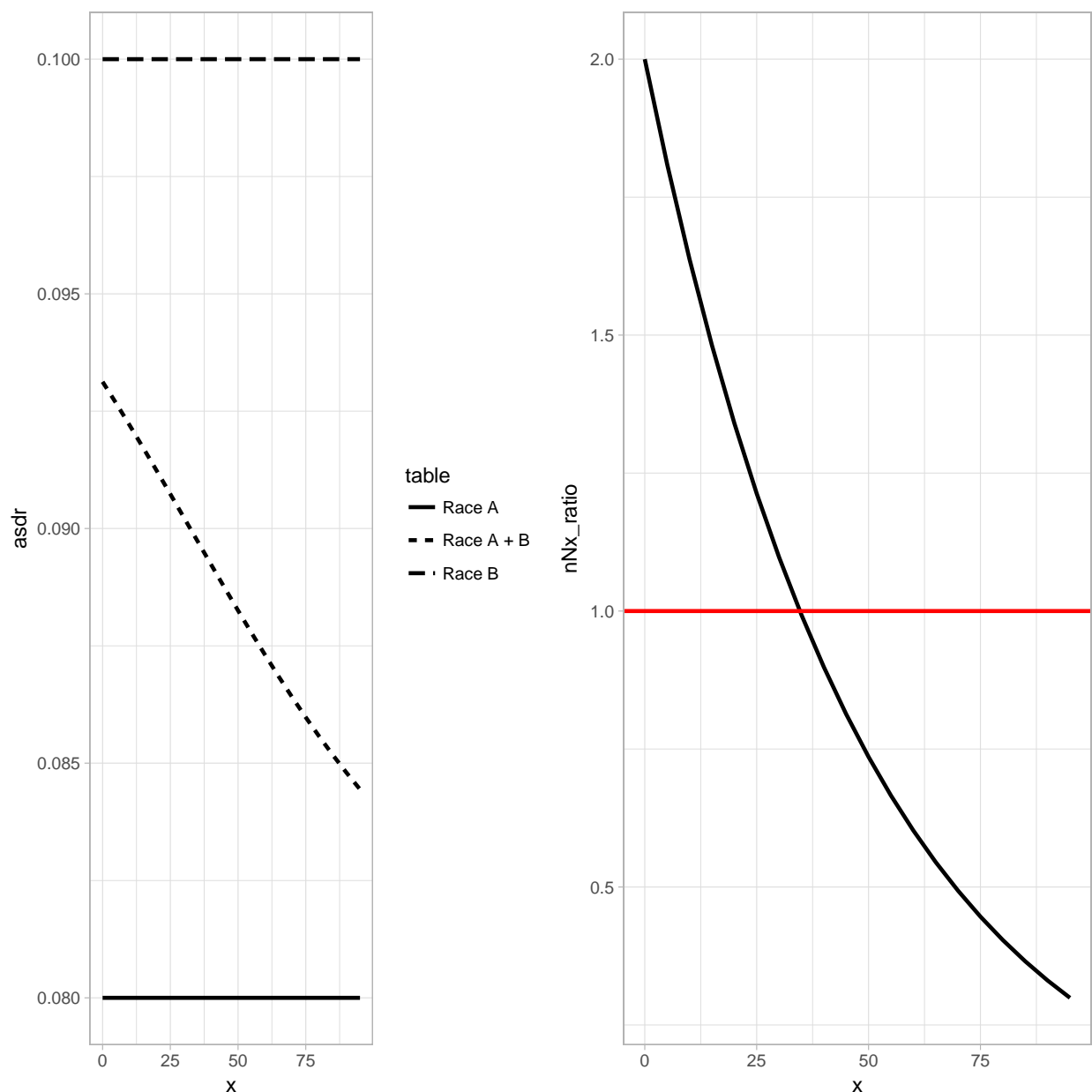
$$\frac{nDx_A + nDx_B}{nNx_A + nNx_B} = \frac{nDx_A + nDx_B}{nNx_A + nNx_B}$$

B.(6)

*# 6. Ratio of nNx in B is higher at first, but then drops and A becomes more influential in the combined calculation
This is due to B starting with more people, but they die off faster than A.*

```
gg1 <- ggplot(data = combined_lt) +
  geom_line(aes(x = x,
                y = asdr,
                linetype = table),
            size = 1) +
  theme_light()

ab_compare <- data.table(x = seq(0, 95, 5),
                        nNx_ratio = lt_b[, 1x] / lt_a[, 1x])
gg2 <- ggplot(data = ab_compare) +
  geom_line(aes(x = x,
                y = nNx_ratio),
            size = 1) +
  geom_hline(aes(yintercept = 1),
             color = 'red',
             size = 1) +
  theme_light()
```



Answer:

As the plot on the left illustrates above, the ASDR is constant over age in both Race A and Race B, but the combined ASDR declines over age. As demonstrated in (5), this combined ASDR is a weighted average of the two separate ASDRs. This decline is due to the structure of the weights changing over age, i.e. the relative population distribution (nNx_A and nNx_B) changing between the two races over age. At birth, there are twice as many people in Race B (10,000) as in group A (5,000). This ratio, $\frac{nNx_B}{nNx_A}$, over age is demonstrated in the plot on the right above. Starting immediately after birth, the relative contribution of Race B to the weighted average ASDR (via nNx_B) begins to decline because the force of mortality is greater in Race B (.10 per year) than Race A (.08 per year). Eventually, the population of Race B drops below the population of Race A somewhere between ages 30 and 35. At this point, the ASDR of Race A is more influential than the ASDR of Race B in the weighted average

ASDR.

B.(6)

Answer:

She is technically correct that, when examined in aggregate, the rate of mortality in the population seems to decline over age. However, as we know Race A and Race B have completely stable mortality rates over age, this conclusion can be viewed as an example of an ecological fallacy. The combined mortality rate in the population declines entirely as a function of the relative population age distributions changing between the races (i.e., the relative structure of the weights changing over age in the weighted-average ASDR). I think this is a disturbing example for interpreting any mortality rate change and deciding if the conclusion is “right” or “wrong”, as the researcher is always susceptible to the ecological fallacy and how they define their population of interest. In this case, we knew race to be a salient category in the population and subset our mortality analysis. However, if we didn’t know “race” was important in this population (or whatever the important underlying stratification in a population may be), we might easily mistake a decline in ASDR as “true” when in fact it is simply an issue of not stratifying properly.