Demographic Methods - Practical 3 (Life Tables I)

2025-10-20

The heading of the R script

The R Script starts by clearing all generated data if any and installing and loading required packages (in this example, lines are commented out using hashtags to prevent execution).

```
rm(list = ls())
#install.packages("stringr")
#install.packages("ggplot2")
#install.packages("tidyverse")
#install.packages("gridExtra")
#library(stringr)
#library(ggplot2)
#library(tidyverse)
#library(gridExtra)
```

Reading the data

The R function "c()" combines values into a vector or list. We can use this function to define the age-specific mortality rates ${}_nM_x$, where x represents the beginning of the age interval, for an unspecified West African population in 1975. The following command lines could be added to the R script.

```
x                   = c(0,1,5,10,15,20,25,30,35,40,45,50,55,60,65,70,75)
nMx                 = c(0.22650,0.03430,0.00198,0.00038,0.00180,0.00252,
                        0.00290,0.00318,0.00352,0.00390,0.00454,0.00490,
                        0.00622,0.00998,0.02066,0.06748,0.31780)
```

Identifying the length of the age intervals n

In R, the function "diff(x,n)" returns the n^{th} difference of x. The first difference of ages x is the length of the age intervals n. The length of the open-ended age interval is undetermined and marked as "NA". A single additional line, like the one shown below, is sufficient to complete this task.

```
n                   = c(diff(x,1),NA)
```

Incorporating assumptions about the distribution of deaths within the age interval, using the function ${}_na_x$

Depending on the source, ${}_na_x$ may represent either a number of years less than n or, alternatively, a fraction of n. Both interpretations are valid, but the chosen definition has implications for the equations used in life table calculations. In this module, ${}_na_x$ is assumed to be half the length of the age interval, except for individuals dying in the first two age intervals. For infant deaths, a value of 0.3 is assumed in contexts of high infant mortality, and 0.1 in contexts of low mortality. In the 1–4 age interval, child deaths are assumed to contribute 0.4 times the length of n. These are coarse assumptions, however, and more precise estimates—as well as a detailed discussion—can be found in: Romero-Prieto, Verhulst, and Guillot (2024). Estimating 1a0 and 4a1 in a life table: A model approach based on newly collected data, *Demography*, 61(3), 643–664. https://doi.org/10.1215/00703370-11330227.

```
nax                 = c(0.3,0.4,rep(0.5, length(nMx) - 3),NA)
```

Note that when defined as a proportion, ${}_na_x$ is undetermined in the open-ended age interval and marked as "NA".

Identifying the open-ended age interval

In R, "!" is used to negate a function or a statement, and "is.na()" returns a selection of NA values. Therefore, "!is.na(n)" returns a selection of age intervals that are not undetermined, as shown by the following line.

```
sEL                 = !is.na(n)
```

"F[sEL]" returns the subset of elements in any vector or list F that satisfy the condition sEL, while "F[!sEL]" returns those that do not satisfy sEL. The condition sEL is useful for identifying the open-ended age interval, **which requires additional assumptions**.

Estimating (conditional) probabilities of dying ${}_nq_x$

Conditional probabilities of dying are estimated from ${}_nM_x$ and ${}_na_x$. Different values of ${}_na_x$ lead to slightly different estimates of ${}_nq_x$. The following line shows how to estimate ${}_nq_x$ for all age intervals except the open-ended one, which is treated separately.

```
nqx                 = n[sEL]*nMx[sEL]/(1 + n[sEL]*(1 - nax[sEL])*nMx[sEL])
```

For the open-ended age interval, it is simply assumed that everyone dies, i.e., a *memento mori* condition to close a life table. Hence, the following line completes the estimation of ${}_\infty q_x$.

```
nqx[!sEL]           = 1
```

Calculating probabilities of surviving ${}_np_x$

Surviving is the opposite event of dying, hence surviving probabilities are calculated as the complement of the conditional probabilities of dying, i.e., ${}_np_x = 1 -_n q_x$, adding the following line to the R script.

```
npx                 = 1 - nqx
```

Defining the radix of a life table and calculating the number of survivors at exact ages l_x

The radix of a life table is used to scale the number of survivors at birth, but could be any (positive) number. The choice of radix does not affect probabilities or the life expectancy, but determines the number of survivors l_x, the number of deaths ${}_nd_x$, the number of person-years at each age interval ${}_nL_x$, and the number of person-years lived above age x, denoted by T_x.

```
radix               = 100000
```

The number of survivors at an exact age l_x, is then calculated from the cumulative product of ${}_np_x$, starting from the radix. The following line complete these calculations at all ages, using the R function "cumprod()".

```
lx                  = cumprod(c(1,npx[sEL]))*radix
```

If the radix is unitary, i.e., $l_0 = 1$, then l_x becomes a probability, and ${}_nL_x$ and T_x quantify years rather than person-years as there is only one person.

Calculating the number of deaths in each age interval ${}_nd_x$

The number of deaths in each age interval is calculated as the product of the number of survivors at the beginning of the age interval and the conditional probability of dying in that interval.

```
ndx                 = lx*nqx
```

One alternative way of calculating ${}_nd_x$ is as the difference between the number of survivors at the beginning and at the end of the age interval, i.e., ${}_nd_x = l_x - l_{x+n}$. This alternative can be implemented in R using the negative of the function "diff()", used before. This alternative calculation produces identical results, except for the open-ended age interval, which is missed as a result of taking the first difference of a vector. Hence, the use of the alternative calculation requires an additional line to complete the calculation of ${}_\infty d_x$ in the open-ended age interval, as shown below.

```
ndx[sEL]            = -diff(lx,1)
ndx[!sEL]           = lx[!sEL]
```

The additional line makes explicit the *memento mori* condition, that everyone dies in the open-ended age interval.

Calculating the number of person-years lived in the age interval ${}_nL_x$

The number of person-years lived in the age interval is calculated as the sum of two components: (i) those who survive to the age interval contribute the length of the age interval, i.e., $n * (l_x -_n d_x)$; and (ii) those who die in the age interval contribute a fraction ${}_na_x$ of the length of the age interval, i.e., $n *_n a_x *_n d_x$. Therefore, ${}_nL_x$ is calculated adding the following line to the R script.

```
nLx                 = n*(lx - ndx) + n*nax*ndx
```

For the open-ended age interval, it is assumed that those who die in that interval live, on average, their life expectancy which is the reciprocal of the age-specific mortality rate, i.e., $1/_\infty M_x$. This assumption is implemented by adding the following line to the R script.

```
nLx[!sEL]           = ndx[!sEL]/nMx[!sEL]
```

Indeed, the open-ended interval should satisfy the following closure conditions: $l_x = {}_\infty d_x$, $T_x = {}_\infty L_x$, and the resulting $e_x = 1/_\infty M_x$.

Calculating the number of person-years lived above age x, denoted by T_x

The number of person-years lived above age x is calculated as the sum of ${}_nL_x$ from age x to the open-ended age interval. In the practice, this is an inverse cumulative sum—from the n^{th} element to the last, that can be implemented in R using the function "cumsum()" with some fixings, as described in the following command line that could be added to the R script.

```
Tx                  = sum(nLx) - (cumsum(nLx) - nLx)
```

An alternative way of calculating T_x is by first reversing the order of ${}_nL_x$ using the R function "rev()", then calculating the cumulative sum of the reversed vector, and finally reversing the order of the resulting vector. This alternative calculation can be implemented in R as follow:

```
Tx                  = rev(cumsum(rev(nLx)))
```

Calculating the life expectancy at age x, denoted by e_x

The life expectancy at age x is calculated as the ratio of T_x to l_x. This is the total number person-years lived above age x, divided by the number of people surviving to that age. Hence, it represents an expected value and is measured in years.

```
ex                  = Tx/lx
```

Consolidating all functions of a life table and extracting results

All functions of a life table can be consolidated in a single table using the R function "data.frame()", as described by the following command line.

```
LT                  = data.frame(x, n, nMx, nax, nqx, npx, lx, ndx, nLx, Tx, ex)
```

The following line could be used to print the full life table, as shown:

```
print(LT)
```

```
##     x  n     nMx nax         nqx       npx        lx        ndx       nLx
## 1   0  1 0.22650 0.3 0.195502999 0.8044970 100000.00 19550.2999  86314.79
## 2   1  4 0.03430 0.4 0.126764728 0.8732353  80449.70 10198.1843 297323.16
## 3   5  5 0.00198 0.5 0.009851236 0.9901488  70251.52   692.0643 349527.42
## 4  10  5 0.00038 0.5 0.001898197 0.9981018  69559.45   132.0375 347467.16
## 5  15  5 0.00180 0.5 0.008959681 0.9910403  69427.41   622.0475 345581.95
## 6  20  5 0.00252 0.5 0.012521117 0.9874789  68805.37   861.5200 341873.03
## 7  25  5 0.00290 0.5 0.014395632 0.9856044  67943.85   978.0946 337274.00
## 8  30  5 0.00318 0.5 0.015774592 0.9842254  66965.75  1056.3574 332187.87
## 9  35  5 0.00352 0.5 0.017446471 0.9825535  65909.39  1149.8863 326672.26
## 10 40  5 0.00390 0.5 0.019311711 0.9806883  64759.51  1250.6169 320671.00
## 11 45  5 0.00454 0.5 0.022445246 0.9775548  63508.89  1425.4727 313980.77
## 12 50  5 0.00490 0.5 0.024203507 0.9757965  62083.42  1502.6365 306660.50
## 13 55  5 0.00622 0.5 0.030623800 0.9693762  60580.78  1855.2137 298265.88
## 14 60  5 0.00998 0.5 0.048685302 0.9513147  58725.57  2859.0720 286480.16
## 15 65  5 0.02066 0.5 0.098226596 0.9017734  55866.50  5487.5758 265613.54
## 16 70  5 0.06748 0.5 0.288696843 0.7113032  50378.92 14544.2353 215534.01
## 17 75 NA 0.31780  NA 1.000000000 0.0000000  35834.69 35834.6852 112758.61
##           Tx        ex
## 1  4884186.1 48.841861
## 2  4797871.3 59.638150
## 3  4500548.2 64.063360
## 4  4151020.7 59.675869
## 5  3803553.6 54.784607
## 6  3457971.6 50.257295
## 7  3116098.6 45.862852
## 8  2778824.6 41.496205
## 9  2446636.7 37.121214
## 10 2119964.5 32.735957
## 11 1799293.5 28.331363
## 12 1485312.7 23.924467
## 13 1178652.2 19.455876
## 14  880386.3 14.991534
## 15  593906.2 10.630811
## 16  328292.6  6.516468
## 17  112758.6  3.146633
```

The following line could be used to print only the age and the life expectancy of the life table, as shown:

```
print(LT[,c("x","ex")])
```

```
##     x        ex
## 1   0 48.841861
## 2   1 59.638150
## 3   5 64.063360
## 4  10 59.675869
```

```
## 5  15 54.784607
## 6  20 50.257295
## 7  25 45.862852
## 8  30 41.496205
## 9  35 37.121214
## 10 40 32.735957
## 11 45 28.331363
## 12 50 23.924467
## 13 55 19.455876
## 14 60 14.991534
## 15 65 10.630811
## 16 70  6.516468
## 17 75  3.146633
```

The following line can be used to print the number of person-years lived in the interval [25, 30), using ${}_nL_x$ as input of the R function "sprintf("format", number)" displaying all digits and one decimal point, as shown:

```
sprintf("%.1f",nLx[x == 25])
```

```
## [1] "337274.0"
```

As a final example, the following line could be used to extract the life expectancy at age 10 with two decimal points from e_x, as shown:

```
sprintf("%.2f",ex[x == 10])
```

```
## [1] "59.68"
```

Exercise 1

Part A. Determine the following quantities:

1. The life expectancy at birth? Hint: Select from the vector e_x the age 0.

```
ex[x == 0]
```

```
## [1] 48.84186
```

2. The life expectancy at age 40?

```
ex[x == 40]
```

```
## [1] 32.73596
```

3. The probability of dying in infancy? Hint: If infancy is defined as the firs year of life, select the probability of dying below age 1.

```
nqx[x == 0]
```

```
## [1] 0.195503
```

An alternative way of calculating that probability is:

```
1 - lx[x == 1]/lx[x == 0]
```

```
## [1] 0.195503
```

4. The number alive (in the life table) at exact age 50? Hint: Return the value the l_x function at age 50.

```
lx[x == 50]
```

```
## [1] 62083.42
```

5. The number of (life table) deaths between exact ages 5 and 10? Hint: There are two approaches: (i) summing the number of deaths across the relevant age intervals; and (ii) differentiating the function l_x at the exact ages involved.

```
sum(ndx[x >= 5 & x < 10])
```

```
## [1] 692.0643
```

```
lx[x == 5] - lx[x == 10]
```

```
## [1] 692.0643
```

Also note that this is a 5-year age interval and we are working with an abridged life table of 5-year age intervals (except for the first five years of life). There is only one value in this life table corresponding to the age interval 5 to 10, i.e., ${}_5d_5$.

```
ndx[x == 5]
```

```
## [1] 692.0643
```

6. The probability of surviving between exact ages 60 and 65? Hint: Report the size of the cohort celebrating their 65^{th} birthday divided by the size of the cohort who celebrated their 60^{th} birthday, five years earlier.

```
lx[x == 65]/lx[x == 60]
```

```
## [1] 0.9513147
```

Considering the age intervals, there is only one value in this life table satisfying this condition, i.e., ${}_5p_{60}$.

```
npx[x == 60]
```

```
## [1] 0.9513147
```

7. The number of years that a newborn is expected to live between exact ages 1 and 5? Hint: You can sum the number of person-years between exact ages 1 and 5, and then divide by the radix of the life table.

```
sum(nLx[x >= 1 & x < 5])/lx[x == 0]              #... either Adding up nLx.#
```

```
## [1] 2.973232
```

```
sum(Tx[x == 1] - Tx[x == 5])/lx[x == 0]          #or decumulating Tx.#
```

```
## [1] 2.973232
```

Note that this expectation could be calculated at a further age, for example age 1.

8. The total number of person-years lived in this life table? Hint: Function T_x quantifies that. Adding up ${}_nL_x$ across all ages could be an alternative solution.

```
Tx[x == 0]
```

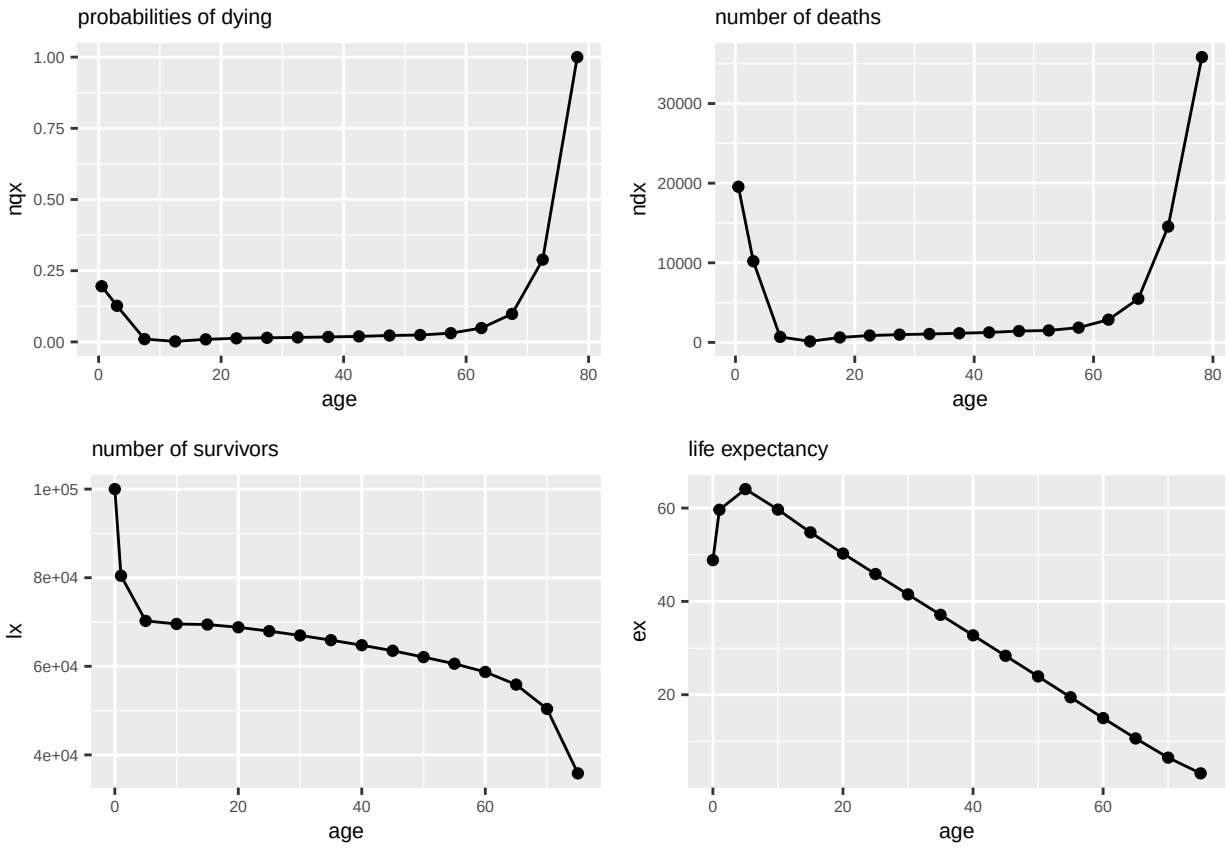
```
## [1] 4884186
```

Part B. Plot ${}_nq_x$, ${}_nd_x$, l_x and e_x against age. Note that some of these quantities pertain to an age group, others to an exact age. Your plots need to reflect that.

Hint: While quantities at exact ages can be plotted at a given value of x, quantities representing age intervals should be plotted at the midpoint of the interval, i.e., $x + n/2$. The only exception is the open-ended age interval, which should be plotted at $x + e_x$.

```r
library(ggplot2)
library(gridExtra)
LT[,"x + n/2"]       = x + n/2                          #Generates a variable "x + n/2", indicating t
LT[!sEL,"x + n/2"]   = x[!sEL] + ex[!sEL]               #As a convention, the expected or average len
Y                    = list("nqx","ndx","lx","ex")      #Defines a list of functions to be plotted.#
X                    = list("x + n/2","x + n/2","x","x") #Accounts for the difference in the x-axis wh
t                    = list("probabilities of dying",
                            "number of deaths",
                            "number of survivors",
                            "life expectancy")          #Generates a list of titles for each function
Fi                   = list()                           #Defines "Fi" as a list of figures as we are
for (i in 1:length(X)) {                                #To repeat the same set of commands per funct
  Fi[[i]] <- local({                                    #Adjudicates to a "local()" all information o
    x  = LT[,X[[i]]]                                    #Extracts from a life table the values of a f
    y  = LT[,Y[[i]]]                                    #Idem.#
    fi = data.frame(x,y)                                #Creates a data frame with the relevant infor

    ggplot(data = fi,aes(x = x, y = y)) +
      geom_point() +
      geom_line() +
      labs(title = t[[i]], x = "age", y = Y[[i]]) +
      theme(plot.title = element_text(size = 8),
            axis.title = element_text(size = 8),
            axis.text = element_text(size = 6))         #Plots each function.#
    })
}
grid.arrange(Fi[[1]],Fi[[2]],Fi[[3]],Fi[[4]], nrow = 2)   #Compiles all plots into one figure.#
```



Part C. What is the probability of dying in the open-ended age interval? Explain your answer.

Hint: sEL selects all ages but the open-ended age interval. !sEL negates the statement sEL and returns just the open-ended age interval. You can use !sEL to select the probability of dying in the open ended-age interval.

```
nqx[!sEL]
```

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

Exercise 2 (Optional)

The life table below is for a high-income country in the 1960s. Fill in the blanks (NA) and interpret the values that you've calculated.

The data for this exercise are extracted from a GitHub repository, using the following command lines.

```
rm(list = ls())
GitHub               = "https://raw.githubusercontent.com/Romero-Prieto/teaching/main/Demographic%20Metl
LT                   = read.csv(GitHub)                    #To pull the data from a GitHub repository.#
print(LT)                                                  #To print data.#
```

```
##     x  n  nax     nqx     lx   ndx      nLx        Tx    ex
## 1   0  1 0.12 0.02450 100000  2450  97844.0        NA 70.10
## 2   1  4 0.40 0.00371  97550   362 389331.2 6912522.6 70.86
## 3   5  5 0.50 0.00218  97188   212 485410.0 6523191.4 67.12
## 4  10  5 0.50 0.00203  96976   197 484387.5 6037781.4 62.26
```

```
## 5  15  5 0.50 0.00474  96779   459 482747.5 5553393.9 57.38
## 6  20  5 0.50 0.00634  96320   611 480072.5        NA 52.64
## 7  25  5 0.50 0.00670  95709   641 476942.5 4590573.9 47.96
## 8  30  5 0.50 0.00844  95068   802 473335.0 4113631.4    NA
## 9  35  5 0.50 0.01208  94266    NA 468482.5 3640296.4 38.62
## 10 40  5 0.50 0.01832  93127  1706 461370.0 3171813.9 34.06
## 11 45  5 0.50 0.02856  91421  2611 450577.5 2710443.9 29.65
## 12 50  5 0.50 0.04474     NA  3973 434117.5 2259866.4 25.45
## 13 55  5 0.50 0.06711  84837  5693       NA 1825748.9 21.52
## 14 60  5 0.50      NA  79144  7761 376317.5 1415796.4 17.89
## 15 65  5 0.50 0.14719  71383 10507 330647.5 1039478.9 14.56
## 16 70  5 0.50 0.20479  60876 12467 273212.5  708831.4 11.64
## 17 75  5 0.50 0.29183  48409 14127 206727.5  435618.9  9.00
## 18 80  5 0.50 0.41990  34282 14395 135422.5  228891.4  6.68
## 19 85 NA   NA 1.00000  19887 19887  93468.9   93468.9  4.70
```

1. Probability of dying between 60 and 64 years

```
x                    = LT[,"x"]                          #To define a variable x, informing the age of
LT[x == 60,"nqx"]    = LT[x == 60,"ndx"]/LT[x == 60,"lx"] #Using "lx" to operate the "lx" column of the
```

${}_5q_{60} = 0.0980618$

2. Number of survivors to the age 50

```
LT[x == 50,"lx"]     = LT[x == 45,"lx"] - LT[x == 45,"ndx"]
```

$l_{50} = 88810$

3. Number of deaths between 35 and 39

```
LT[x == 35,"ndx"]    = LT[x == 35,"lx"] - LT[x == 40,"lx"]
```

${}_5d_{35} = 1139$

4. Number of person-years lived between 55 and 59

```
LT[x == 55,"nLx"]    = LT[x == 55,"Tx"] - LT[x == 60,"Tx"]
```

${}_5L_{55} = 409952.5$

5. Number of person-years lived above age 0

```
LT[x == 0,"Tx"]      = LT[x == 0,"nLx"] + LT[x == 1,"Tx"]
```

$T_0 = 7010366.6$

6. Number of person-years lived above age 20

```
LT[x == 20,"Tx"]     = LT[x == 20,"nLx"] + LT[x == 25,"Tx"]
```

$T_{20} = 5070646.4$

7. Life expectancy at age 30

```
LT[x == 30,"ex"]     = LT[x == 30,"Tx"]/LT[x == 30,"lx"]
```

$e_{30} = 43.2704107$

Exercise 3 (Optional)

Give definitions/interpretations of the following life table expressions:

1. l_5: number of survivors at exact age 5.
2. ${}_5q_{15}$: probability of dying between exact ages 15 and 20.
3. ${}_{45}q_{15}$: probability of dying between exact ages 15 and 60.
4. T_{30}: total number of person-years lived above exact age 30.
5. ${}_5d_{60}$: estimated number of deaths between exact ages 60 and 65 in a life table.
6. ${}_5D_{60}$: observed number of deaths between exact ages 60 and 65 in a population.
7. ${}_4m_1$: age-specific mortality rate between exact ages 1 and 5.
8. e_{60}: life expectancy at exact age 60.
9. ${}_{30}L_5$: number of person-years lived between exact ages 5 and 35.
10. $l_{20}/radix$: probability to survive to age 20.
11. l_{20}/l_5: probability to survive between exact ages 5 and 20.
12. ${}_5d_{50}/l_0$: $({}_5d_{50}/l_{50})(l_{50}/l_0)$ probability that a newborn dies between exact ages 50 and 55.

Exercise 4 (Optional)

Can you think of one or more ways to improve the life table in exercise 1 (and thus obtain more accurate life expectancy estimates)?

Graduation of ${}_na_x$ values. Smaller age intervals. Extending the life table to have an open-ended age interval at age 100.