

# Life expectancy: a global health index

2024-09-05

- M1 MIDS MA7BY020
- [Université Paris Cité](#)
- Année 2024-2025
- [Course Homepage](#)
- [Moodle](#)



## Objectives

## Loading

```
datafile <- 'tamed_life_table.Rds'
fpath <- str_c("./DATA/", datafile) # here::here('DATA', datafile) # check getwd() if pr

if (! file.exists(fpath)) {
  download.file("https://stephane-v-boucheron.fr/data/tamed_life_table.Rds",
               fpath,
               mode="wb")
}

life_table <- readr::read_rds(fpath)
```

### **i** References

For definitions of column, check on <http://www.mortality.org> the meaning of the different columns.

See also *Demography: Measuring and Modeling Population Processes* by SH Preston, P Heuveline, and M Guillot. Blackwell. Oxford. 2001.

Document [Tables de mortalité françaises pour les XIXe et XXe siècles et projections pour le XXIe siècle](#) contains detailed information on the construction of Life Tables for France.

In the sequel, we denote by  $F_t$  the *cumulative distribution function* for year  $t$ . We agree on  $\bar{F}_t = 1 - F_t$  and  $F_t(-1) = 0$ . Henceforth,  $\bar{F}$  is called the *survival* function.

**qx** (age-specific) risk of death at age  $x$ , or mortality quotient at given age  $x$  for given year  $t$ .

**i** About the definition of  $q_{t,x}$ 

Defining and computing  $q_{t,x}$  does not boil down to knowing the number of people at age  $x$  at the beginning of year  $t$  and knowing how many of them died during year  $t$ . If we want to be rigorous, we need to know all life lines in the Lexis diagram, or equivalently, how many people at Age  $x$  were alive on each day of Year  $t$ .

**🔥** Mortality quotients define a probability distribution

For a given year  $t$ , the sequence of mortality quotients define a survival function  $\bar{F}_t$  using the following recursion:

$$q_{t,x} = \frac{\bar{F}_t(x) - \bar{F}_t(x+1)}{\bar{F}_t(x)}$$

with boundary condition  $\bar{F}_t(-1) = 1$ .

This recursion can also be read as:

$$\bar{F}_t(x+1) = \bar{F}_t(x) \times (1 - q_{t,x+1}).$$

This artificial probability distribution is used to define and compute life expectancies.



$q_{t,x}$  is the *hazard rate* of  $\bar{F}_t$  at age  $x$ .

**ex:** Residual Life Expectancy at age  $x$  and year  $t$

This is the expectation of  $X - x$  for a random variable  $X$  distributed according to  $\bar{F}_t$  conditionally on the event  $\{X \geq x\}$ . That is  $e_{t,x}$  is the expectation of the probability distribution defined by  $\bar{F}_t(\cdot + x - 1) / \bar{F}_t(x - 1)$ .

## Rearrangement

**i Question**

From dataframe `life_table`, compute another dataframe called `life_table_pivot` with primary key `Country`, `Gender` and `Year`, with a column for each `Age` from 0 up to 110. For each age column, the entry should be the central death rate at the age defined by column, for `Country`, `Gender` and `Year` identifying the row.

You may use functions `pivot_wider`, `pivot_longer` from `tidyr::` package.

The resulting schema should look like:

Column Name	Type
Country	factor
Gender	factor
Year	integer
0	double
1	double
2	double
3	double
:	:

```
life_table_pivot <- life_table |>
  select(Country, Gender, Year, Age, qx) |>
  pivot_wider(names_from = "Age",
              values_from = "qx")
```

**i Question**

Using `life_table_pivot` compute life expectancy at birth for each `Country`, `Gender` and `Year` using formula

$$e_{t,0} = \sum_{x=0}^{\infty} \bar{F}_t(x)$$

```
tmp <- life_table_pivot |>
  select(-c(Country, Gender, Year)) |>
  as.matrix()

lex <- apply(1- tmp,
            MARGIN = 1,
            FUN = function(x) {sum(cumprod(x))}
)

rm(tmp)

lex_table<- life_table_pivot |>
  add_column(lex=round(lex, 2)) |>
  select(Country, Year, Gender, lex)

rm(lex)

small_lex_table <- lex_table |>
  sample_n(size= 10) |>
  arrange(Year)
```

```
gt_1 <- small_lex_table |>
  gt() |>
  tab_header(
    title = "Computed Life Expectancies at birth",
    subtitle = "a sample"
  )
```

```
gt_1 |>
  as_latex()
```

## Life expectancy and window functions

### i Question

Write a function that takes as input a vector of mortality quotients, as well as an age, and returns the residual life expectancy corresponding to the vector and the given age.

```
ex <- function(qx){
  sum(cumprod(1 - qx))
}
```

### i Question

Write a function that takes as input a dataframe with the same schema as `life_table` and returns a data frame with columns `Country`, `Gender`, `Year`, `Age` defining a primary key and a column `res_lex` containing *residual life expectancy* corresponding to the primary key.

In order to compute residual life expectancies, you may consider using `window` functions over appropriately defined windows. The next window function suffices to compute life expectancy at birth. It computes the logarithm of survival probabilities for each `Country`, `Year`, `Gender` (partition) at each `Age`. Note that the expression mentions an aggregation function `sum` and that the correction of the result is ensured by a correct design of the `frame` argument.

```
df <- life_table |>
  select(Country, Gender, Year, Age, qx) |>
  group_by(Country, Year, Gender) |>
  arrange(Age) |>
  mutate(sx = cumprod(1-qx)) # window function
```

```
df_leb <- df |>
  summarise(e_0 = sum(sx)) |> # aggregation function
  ungroup()
```

```
gt_2 <- df_leb |>
  sample_n(size = 20) |>
  arrange(Year) |>
  gt() |>
  tab_header(
    title = "Computed Life Expectancies at birth",
    subtitle = "a sample"
  )
```

```
gt_2 |>
  as_latex()
```

Computation of Life Expectancy at birth boils down to compute survival probabilities using one window function, grouping with respect to country, gender and year, ordering by age and summing up survival probabilities.

### Question

Compute residual life expectancies at all ages using window functions  
You can use `slider::slide()`.

In order to compute *Residual Life Expectancies* at all ages, instead of performing aggregation, we compute a second window function.


The window is more sophisticated than the previous one, we still need partitioning by **Year**, **Country** and **Gender**, ordering by **Age**, but we also need to sum over conditional survival probabilities, which are just ratios of survival probabilities, but over a *frame* defined by the current **Age** and all ages above.

```
df_ex <- df |>
  mutate(ex = slide_dbl(sx, ~ sum(.x), .before=0, .after=Inf)/lag(sx, default=1)) |>
  ungroup()
```

### 

This is slow.

## Computing residual life expectancies using window functions and accumulate

-  The official calculation of residual life expectancies assumes that except at age 0 and great age, people die uniformly at random between age  $x$  and  $x + 1$ :

$$e_{t,x} = (1 - q_{t,x}) \times (1 + e_{t,x+1}) + \frac{1}{2} \times q_{t,x}$$

This recursion suggests a more efficient to compute *residual life expectancies* at all ages.

Indeed, `purrr::accumulate()` allows to compute all values for  $e_{t,x}$  using exactly one pass over the table.

See <https://purrr.tidyverse.org/reference/accumulate.html>

### Question

```
#' Compute residual life expectancies from mortality quotients
#'
#' @param z a vector of mortality quotients ordered by decreasing ages
#'
#' @return a vector of residual life expectancies ordered by decreasing ages
#' @export
#'
```

```
#' @examples
#' rlex(rep(.1, 10)) # constant haszard rate
#' rlex(dpois(9:0, 1)/c(ppois(8:0, 1, lower.tail = F),1)) # increasing hazard rate
rlex <- function(z) {
  purrr::accumulate(z,
    .f= function(x, y){(1 - y) * (1 + x) + y/2},
    .init= 0)[-1]
}
```

**i** Documenting a function[See R Packages Book](#)**i** Question

```
df_rle <- life_table |>
  select(Country, Gender, Year, Age, qx) |>
  group_by(Country, Year, Gender) |>
  arrange(desc(Age)) |>
  mutate(`Residual Life Expectancy`=rlex(qx)) |>
  ungroup()
```

```
# tbl_ResLifeExpectancy |>
#
fn_1 <- function(df, up_a=10, g= 'Female', y=2016) {
  df |>
  filter(Age < up_a,
    Gender == g,
    Year == y) |>
  select(Country, Age, `Residual Life Expectancy`) |>
  arrange(Age) |>
  pivot_wider(
    names_from=Age,
    values_from= `Residual Life Expectancy`
  ) |>
  gt() |>
  tab_header(
    title = "Computed Life Expectancies at different ages",
    subtitle = glue::glue("{g} under {up_a} for Year {y}")
  ) |>
  fmt_engineering(columns=-Country) |>
  gt::tab_spanner(label = "Age", columns = seq(2,1+up_a))
}
```

```
gt_3 <- fn_1(df_rle)
```

```
gt_3 |>
  as_latex()
```

**i Question**

Compute and display residual life expectancies for ages 0 to 9 for year 1972

```
gt_3_b <- fn_1(df_rle, y=1972)
```

```
gt_3_b |>  
  as_latex()
```

Note that for year 1972, except in the Netherlands and in Sweeden, for girls, residual life expectancies at age 0 are slightly lower than residual life expectancies at age 1.

Is it a surprise?

**i Question**

Plot residual life expectancy as a function of **Year** at ages 60 and 65, facet by **Gender** and **Country**.

```
df_tmp <- with(params,  
  df_rle |>  
    filter(Age %in% c(60, 65),  
      between(Year, year_p, year_e),  
      Gender %in% c('Male', 'Female')  
    )  
)
```

```
df_tmp |>  
  ggplot() +  
  aes(x=Year, y=ex, group=Age, color=as_factor(Age)) +  
  labs(color="Age") +  
  ylab("Residual life expectancy") +  
  geom_line() +  
  facet_grid(rows=vars(Country), cols=vars(Gender)) +  
  ggtitle("Evolution of residual life expectancy at 60 and 65")
```

**i Comment.**

Except in Spain, Residual Life Expectancies started to take off late, after 1970.

**i Question**

```
df_pension <- with(params,  
  df_rle |>  
    filter(between(Age, 60, 90),  
      between(Year, year_p, year_e),  
      Gender %in% c('Male', 'Female'))  
)
```

```
{  
  df_pension |>  
  filter(Year %% 10 == 0) |>  
  ggplot() +
```

```
aes(y=ex, x=Age, linetype=Country, col=Country) +  
labs(linetype="Country", col="Country") +  
ylab("Residual life expectancy") +  
geom_line(size=.2) +  
facet_grid(cols=vars(Gender), rows=vars(Year)) +  
ggtitle("Residual Life Expectancy takes off") +  
theme_minimal()  
}
```