# Homework 2021: SVD analysis & Life Tables

## Deadline: 2021-12-09


### Name1 and Name2


### 2021-11-09



# 1 Deliverables

- **2021-12-09:** a `nom1_nom2.Rmd` file that can be `knitted` under `rstudio`
  - Your file should be knitted without errors and the result should be an html document that can be viewed in a modern browser.
  - The file should contain the code used to generate plots and numerical summaries
  - The file should contain the texts of your comments (either in English or in French)
  - Comments should be written with care, precision and should be concise
  - File `nom1_nom2.Rmd` will be uploaded on `Moodle`
- **2021-12-14:** a 20 minutes oral presentation a some material extracted from your first deliverable `nom1_nom2.Rmd`
  - The presentation consists of a 12 minutes talk and 8 minutes of questions and answers
  - The presentation is supported by slides
  - Slides generated from `Rmarkdown` should be favored. You may choose format you prefer ( `binb` , `xaringan` , `slidy` , `ioslides` , …)

---

# 2 Objectives

This notebook aims at

- working with **tables** ( `data.frames` , `tibbles` , `data.tables` , …) using `dplyr` or any other query language (as provided for example by `data.table` )
- visualizing demographic data as provided by Human Mortality Database organization (https://www.mortality.org).
- using **PCA** and other matrix oriented methods to explore a multivariate datasets (lifetables may be considered as multivariate datasets)

# 3 Life tables data (ETL)

Life data tables have been downloaded from https://www.mortality.org (https://www.mortality.org). They have been worked our for you and can be downloaded from URL 'https://www.dropbox.com/s/tnci38tqchxwic6/full_life_table.Rds?dl=0 (https://www.dropbox.com/s/tnci38tqchxwic6/full_life_table.Rds?dl=0)', saved in your working directory.

If you install and load package https://cran.r-project.org/web/packages/demography/index.html (https://cran.r-project.org/web/packages/demography/index.html), you will also find life data tables.

We investigate life tables describing countries from Western Europe (France, Great Britain –actually England and Wales–, Italy, the Netherlands, Spain, and Sweden) and the United States.

We load the one-year lifetables for female, male and whole population for the different countries.

Download data from https://www.dropbox.com/s/tnci38tqchxwic6/full_life_table.Rds?dl=0 (https://www.dropbox.com/s/tnci38tqchxwic6/full_life_table.Rds?dl=0) in your *working* directory. Save it as `full_life_table.Rds` . Load it in memory using `readr::read_rds()` .

```
fpath <- 'full_life_table.Rds'  # once you have downloaded the file

if (! file.exists(fpath)){
  cat(glue('{fpath} should be in working directory!'))
} else {
  life_table <- readr::read_rds(fpath)
  glimpse(life_table)
}
```

Check on http://www.mortality.org (http://www.mortality.org) the meaning of the different columns:

Document Tables de mortalité françaises pour les XIXe et XXe siècles et projections pour le XXIe siècle (https://www.lifetable.de/data/FRA/FRA000018061997CY1.pdf) contains detailed information on the construction of Life Tables for France. Two kinds of Life Tables can be distinguished: *Table du moment* which contain for each calendar year, the mortality risks at different ages for that very year; and *Tables de génération* which contain for a given year of birth, the mortality risks at which an individual born during that year has been exposed.

The life tables investigated in this homework are *Table du moment*. According to the document by Vallin and Meslé, building the life tables required decisions and doctoring.

See (among other things)

- p. 19 Abrupt changes in mortality quotients at some ages for a given calendar year
- Estimating mortality quotients at great age.

Have a look at Lexis diagram (https://en.wikipedia.org/wiki/Lexis_diagram).

---

- [ ] Henceforth, the universal table is named `life_table` , its schema should be the following.

The life tables you will be working on are called *period life tables*.

| **Column Name** | **Column Type** | **Meaning** |
|---|---|---|
| **Year** | integer | |
| **Age** | integer | Age $x$ |
| mx | double | Central death rate at age $x$: $m_x$ |
| **qx** | double | Probability of dying between the ages of $x$ and age $x+1$ $q_x = \frac{m_x}{1+m_x/2}$ |
| ax | double | $.5$ except at age $0$ |
| lx | integer | Number of persons still alive at age $x$ in a fictitious cohort of $100000$ |
| dx | integer | Number of persons deceased between age $x$ and $x+1$ during year in fictitious cohort |

| Column Name | Column Type | Meaning |
|---|---|---|
| Lx | integer | Number of pearson-years lived from age $x$ to $x+1$, $L_x = \ell_x - d_x \times a_x$ in fictitious cohort |
| Tx | integer | |
| ex | double | Residual life expectancy at age $x$ |
| **Country** | factor | Netherlands/… |
| **Gender** | factor | Female/Male |

See Preston *et al.* for details and explanations.

# 4 Western countries in 1948

- [ ] Plot the *mortality quotients* of all Countries at all ages for year 1948.
- [ ] Comment
- [ ] Plot ratios between *mortality quotients* in European countries and *mortality quotients* in the USA in 1948.
- [ ] Comment

# 5 Death rates evolution since WW II

- [ ] Plot *mortality quotients* (column `qx` ) for both genders as a function of `Age` for years `1946, 1956, ...` up to `2016` . Use aesthetics to distinguish years.

- [ ] Facet by `Gender` and `Country`

- [ ] Write a function `ratio_mortality_rates` with signature `function(df, reference_year=1946, target_years=seq(1946, 2016, 10))` that takes as input:

    - a dataframe with the same schema as `life_table` ,
    - a reference year `ref_year` and
    - a sequence of years `target_years`

and that returns a dataframe with schema:

| Column Name | Column Type |
|---|---|
| Year | integer |
| Age | integer |
| qx | double |
| qx.ref_year | double |
| Country | factor |
| Gender | factor |

where `(Country, Year, Age, Gender)` serves as a *primary key*, `qx` denotes the mortality quotient at `Age` for `Year` and `Gender` in `Country` whereas `qx_ref_year` denotes mortality quotient at `Age` for argument `reference_year` in `Country` for `Gender` .

- [ ] Draw plots displaying the ratio $q_{x,t}/q_{x,1946}$ for ages $x \in 1, \ldots, 90$ and year $t$ for $t \in 1946, \ldots, 2016$ where $q_{x,t}$ is the mortality quotient at age $x$ during year $t$.

    1. Handle both genders and countries `Spain` , `Italy` , `France` , `England & Wales` , `USA` , `Sweden` , `Netherlands` .

2. One properly facetted plot is enough.

- [ ] Comment

# 6 Trends

- [ ] Plot mortality quotients at ages $0, 1, 5$ as a function of time. Facet by Gender and Country

- [ ] Comment

- [ ] Plot mortality quotients at ages $15, 20, 40, 60$ as a function of time. Facet by `Gender` and `Country`

- [ ] Comment

# 7 Rearrangement

- [ ] 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 mortality quotient 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 |
| ⋮ | ⋮ |

- [ ] Using `life_table_pivot` compute life expectancy at birth for each Country, Gender and Year

# 8 Life expectancy

- [ ] 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.

- [ ] 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 pimary key.

In order to compute residual life expectancies, you may consider using `window` functions oer apropriately defined windows. Package `dplyr` does not offer a rich API for window functions. Package `dbplyr` does.

- [ ] Plot residual life expectancy as a function of `Year` at ages $60$ and $65$, facet by `Gender` and `Country`.

# 9 PCA and SVD over log-mortality tables

- [ ] Pick a Country, a Gender, a range of years `1948:2010`. Extract the corresponding lines from `life_table_pivot`. Take *logarithms* of mortality quaotients and perform principal component analysis. Assess the results of PCA with and without centering and standardizing the columns
- [ ] Comment the screeplot(s)

- [ ] Comment the correlation circle(s)
- [ ] Comment the biplot(s)
- [ ] Choose the combination of centering and scaling you find the most relevant. Motivate your choice
- [ ] Perform PCA on all countries and genders with the chosen combination of centering and scaling
- [ ] Combine the screeplots for different countries (for each gender). Comment

# 10 Lee-Carter model for US mortality

During the last century, in the USA and in western Europe, mortality quotients at all ages have exhibited a general decreasing trend. This decreasing trend has not always been homogeneous across ages.

The Lee-Carter model has been designed to model and forecast the evolution of the log-mortality quotients for the United States during the XXth century.

Let $A_{x,t}$ denote the log mortality quotient at age $x$ during year $t \in T$ for a given population (defined by Gender and Country).

The Lee-Carter model assumes that observed loagrithmic mortality quotients are sampled according to the following model

$$A_{x,t} \sim_{\text{independent}} a_x + b_x \kappa_t + \epsilon_{x,t}$$

where $(a_x)_x, (b_x)_x$ and $(\kappa_t)_t$ are unknown vectors that satisfy

$$a_x = \frac{1}{|T|} \sum_{t \in T} A_{x,t} \qquad \sum_{t \in T} \kappa_t = 0 \qquad \sum_x b_x^2 = 1$$

and $\epsilon_{x,t}$ are i.i.d Gaussian random variables.

## 10.1 US data

- [ ] Fit a Lee-Carter model on the American data (for Male and Female data) training on years `1933` up to `1995` .
- [ ] Compare the fit provided by the Lee-Carter model with the fit provided by a rank $2$ truncated SVD
- [ ] Compare vectors avec $(a_x)_x, (b_x)_x$ and $(\kappa_t)_t$ with appropriate singular vectors.
- [ ] Use the Lee-Carter model to predict the mortality quotients for years $2000$ up to $2015$
- [ ] Plot predictions and observations for years $2000, 2005, 2010, 2015$

## 10.2 Application of Lee-Carter model to a European Country

- [ ] Fit a Lee-Carter model to a European country
- [ ] Comment
- [ ] Compare with rank-2 truncated SVD
- [ ] Use the Lee-Carter model to predict the mortality quotients for years $2000$ up to $2015$ Plot predictions and observations for years $2000, 2005, 2010, 2015$

## 10.3 Predictions of life expectancies at different ages

- [ ] Use Lee-Carter approximation to approximate residual life expectations
- [ ] Compare with observed residual life expectations

# 11 References

**Life tables and demography**


- Human Mortality Database (https://www.mortality.org)

- Tables de mortalité françaises, Jacques Vallin et France Meslé (https://www.lifetable.de/data/FRA/FRA000018061997CY1.pdf)
- [Modeling and Forecasting U.S. Mortality, R.D.Lee and L.R. Carter, JASA 1992]
- [Les dimensions de la mortalité, S. Ledermann, Jean Breas, Population, 1959]
- Murphy, M. (2001). Samuel H. Preston, Patrick Heuveline and Michel Guillot, Demography: Measuring and Modeling Population Processes.

**Graphics and reporting**

- Interactive web-based data visualization with R, plotly, and shiny (https://plotly-r.com/index.html)
- R for Data Science (https://r4ds.had.co.nz)
- Layered graphics (http://vita.had.co.nz/papers/layered-grammar.pdf)
- Plotly (http://plotly.com/)

**Tidyverse**

- tidyselect (https://tidyselect.r-lib.org/articles/tidyselect.html)
- dbplyr (https://cran.r-project.org/web/packages/dbplyr/vignettes/dbplyr.html)
- data.table (https://github.com/Rdatatable/data.table)
- DT (https://rstudio.github.io/DT/)

**PCA, SVD**

- FactoMineR (http://factominer.free.fr/index_fr.html)
- ade4 (http://pbil.univ-lyon1.fr/ade4/accueil.php)
- FactoInvestigate (http://factominer.free.fr/reporting/index_fr.html)
- PCA and Tidyverse (https://cmdlinetips.com/2019/05/how-to-do-pca-in-tidyverse-framework/)
- tidyprcomp (https://broom.tidyverse.org/reference/tidy.prcomp.html)

**Demography**

- R package demography (https://cran.r-project.org/web/packages/demography/demography.pdf)