KOSTAT-UNFPA Summer Seminar on Population

Workshop 1. Introduction to Demography

Day 4: Population Growth

Instructor:
Tim Riffe `tim.riffe@ehu.eus`

Assistant:
Inchan Hwang `inchan@yonsei.ac.kr`

26 June 2025

Contents

1   Summary

Today we cover classic concepts of population growth, based on assumptions of constant fertility and mortality dynamics. Note the concepts around population growth or reproductivity discussed here are often misinterpreted and misused for nationalist and racist agendas. We're smarter than that and understand that these are just partially useful models. These are conceptual models, not forecasts, and many simplifying assumptions are applied when modeling population growth. As usual we will use tidy coding where possible, but this isn't always possible for things that are iterative or involve matrix algebra, so I'll need to introduce new programming concepts today.

2   Data

We will use the Spanish data from the HMD (Human Mortality Database 2018) and HFD (Human Fertility Database 2018) today. Please load the following datasets (I iwll show you how to load datasets straight from github in class!)

```r
#setwd()
library(tidyverse)
library(readr)
#Birth counts in 2014
B2014 <- read_csv("Data/BirthsSpain2014.txt",
                  show_col_types = FALSE)
# Death counts in 2014
D2014 <-
  read_csv("Data/DeathsSpain2014.txt",
           show_col_types = FALSE)

# Population counts on January 1st, 2014
P2014 <- read_csv("Data/PopulationSpain2014.txt",
                  show_col_types = FALSE)

# Population counts on January 1st, 2015
P2015 <- read_csv("Data/PopulationSpain2015.txt",
                  show_col_types = FALSE)

ES2014 <-
  P2014 |>
  bind_rows(P2015) |>
  pivot_wider(names_from = year,
              values_from = population,
              names_prefix = "P") |>
  left_join(D2014, by = c("age", "sex")) |>
  left_join(B2014, by = c("age")) |>
  arrange(sex, age) |>
  mutate(births = if_else(is.na(births),0,births))
#Births in 2014

# 2014 Life table
LT_url <- "https://raw.githubusercontent.com/timriffe/KOSTAT_Workshop1/master/Data/LTSpain20
LT <- read_delim(LT_url,
                 delim = " ",
                 show_col_types = FALSE) |>
     rename(age = Age)
```

3 Crude growth

3.1 Balancing equation

Changes in population size are affected by changes in the number of births, deaths and migration.

- Two ways of entry: Birth and Immigration
- Two ways of exit: Death and Emigration

As there are only two ways of entry and two ways of exit, changes in population size come from changes in the magnitude of these flows:

$$N(T) = N(0) + B[0,T] - D[0,T] + I[0,T] - E[0,T]$$

where

- $N(T)$ and $N(0)$ are the number of persons alive at time T and 0, respectively
- $B[0,t]$ is the number of births between times 0 and T
- $D[0,t]$ is the number of deaths between times 0 and T
- $I[0,t]$ is the number of immigrations between times 0 and T
- $E[0,t]$ is the number of emigrations between times 0 and T

3.2 Crude growth rate

The crude growth rate is the change in population size relative to the person-year in a given time interval (e.g. from year 0 to T).

Given the balancing equation, the crude growth rate (CGR) of a population is:

$$\frac{N(T) - N(0)}{PY[0,T]} = \frac{B[0,T]}{PY[0,T]} - \frac{D[0,T]}{PY[0,T]} + \frac{I[0,T]}{PY[0,T]} - \frac{E[0,T]}{PY[0,T]}$$

$$CGR[0,T] = \underbrace{CBR[0,T] - CDR[0,T]}_{CRNI[0,T]} + \underbrace{CIR[0,T] - CER[0,T]}_{CRNM_[0,T]}$$

- $CBR[0,T]$ is the crude birth rate
- $CDR[0,T]$ is the crude death rate
- $CIR[0,T]$ is the crude immigration rate
- $CER[0,T]$ is the crude emigration rate
- $CRNI[0,T]$ is the crude rate of natural increase
- $CRNM[0,T]$ crude rate of net migration

We don't have information here on in and out migration, but we can calculate the other crude rates.

```
Components <-
  ES2014 |>
  filter(sex == "Total") |>
  summarize(P1 = sum(P2014),
            P2 = sum(P2015),
            B = sum(births, na.rm = TRUE),
            D  = sum(deaths),
            )|>
  mutate(PY = (P1 + P2) / 2,
         NMig = P2 - (P1 + B - D))

CrudeRates <-
Components %>%
  mutate(CGR = (P2 - P1) / PY,
         CBR = B / PY,
         CDR = D / PY,
```

```
        CRNI = CBR - CDR,
        CRNM = NMig / PY,
        .keep = "none")

CrudeRates
```

```
## # A tibble: 1 x 5
##       CGR      CBR      CDR      CRNI      CRNM
##     <dbl>    <dbl>    <dbl>     <dbl>     <dbl>
## 1 -0.00135 0.00917 0.00847 0.000696 -0.00205
```

The CGR suggests a decline of Spain's population size between 2014 and 2015. However, Spain has recorded a positive CRNI. A possible explanation is that there was more out-migration than in-migration in Spain in 2014, reducing the population size and opposing the natural increase. This residual-based method of estimating net migration is often susceptible to errors in the data components on which it is based.

3.3   Geometric growth

Different growth rates can be assumed in a population over time. If we consider a geometric growth over time, the annual rate of change (r) in a population at time 0 and t is equal to:

$$r = \frac{N(T) - N(0)}{N(0)} \frac{1}{T}$$

Thus, if r is constant over time the population size at time t can be found by

$$N(t) = N(0)[1 + r]^t$$

```
geometric_growth <-
  Components |>
  mutate(r_geom = (P2 - P1) / P1,   # same as CBR here
         r_nat = (B - D) / P1)  |>  # same as CRNI here
  cross_join(tibble(t = 0:10)) |>
  mutate(Pt_geom = P1 * (1 + r_geom) ^ t,
         Pt_nat = P1 * (1 + r_nat) ^ t) |>
  select(t, Pt_geom, Pt_nat)

geometric_growth %>%
  pivot_longer(-t, names_to = "type", values_to = "projection") %>%
  mutate(projection = projection / 1e6) %>%
  ggplot(aes(x = t, y = projection, color = type)) +
  geom_line() +
  xlab("projection (million)") +
    labs(title = "Spanish population under geometric growth") +
  theme_minimal()
```

Spanish population under geometric growth
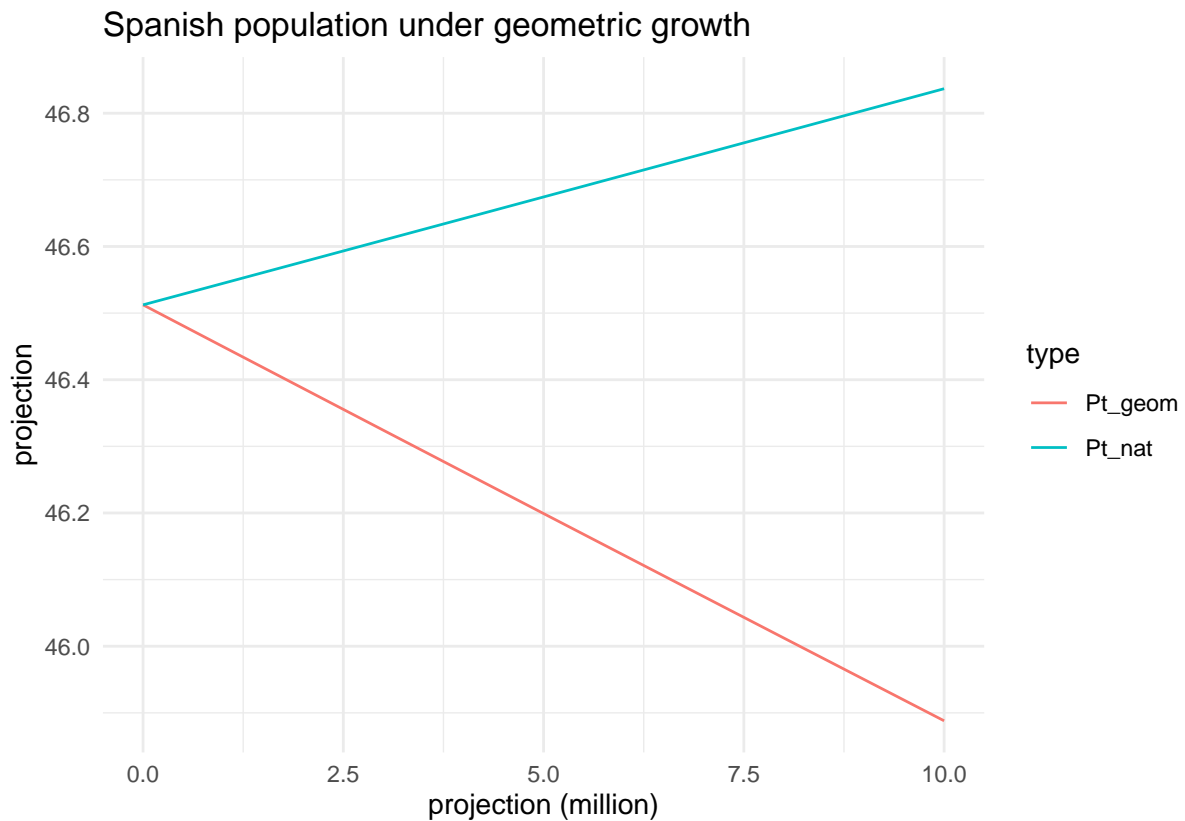
3.4 Exponential growth

The exponential growth can simply be seen as a continuous version of the geometric growth. Here, r is the instantaneous growth rate. If we consider an exponential growth over time, the annual rate of change (r) in a population at time 0 and t is equal to:

$$r = \frac{ln\left(\frac{N(T)}{N(0)}\right)}{T}$$

Thus, if r is constant over time the population size at time t can be found by

$$N(t) = N(0)e^{r*t}$$

```r
# Exponential growth
exponential_growth <-
  Components %>%
  mutate(r_exp = log(P2 / P1),
         r_nat_exp = log((P1 + B - D) / P1)) %>%
  cross_join(tibble(t = 0:10)) %>%
  mutate(Pt_exp = P1 * exp(r_exp * t),
         Pt_nat_exp = P1 * exp(r_nat_exp * t)) %>%
  select(t, Pt_exp, Pt_nat_exp, r_exp, r_nat_exp)

#It we assume a close population (no migration)

exponential_growth %>%
  select(-r_exp, -r_nat_exp) %>%
```
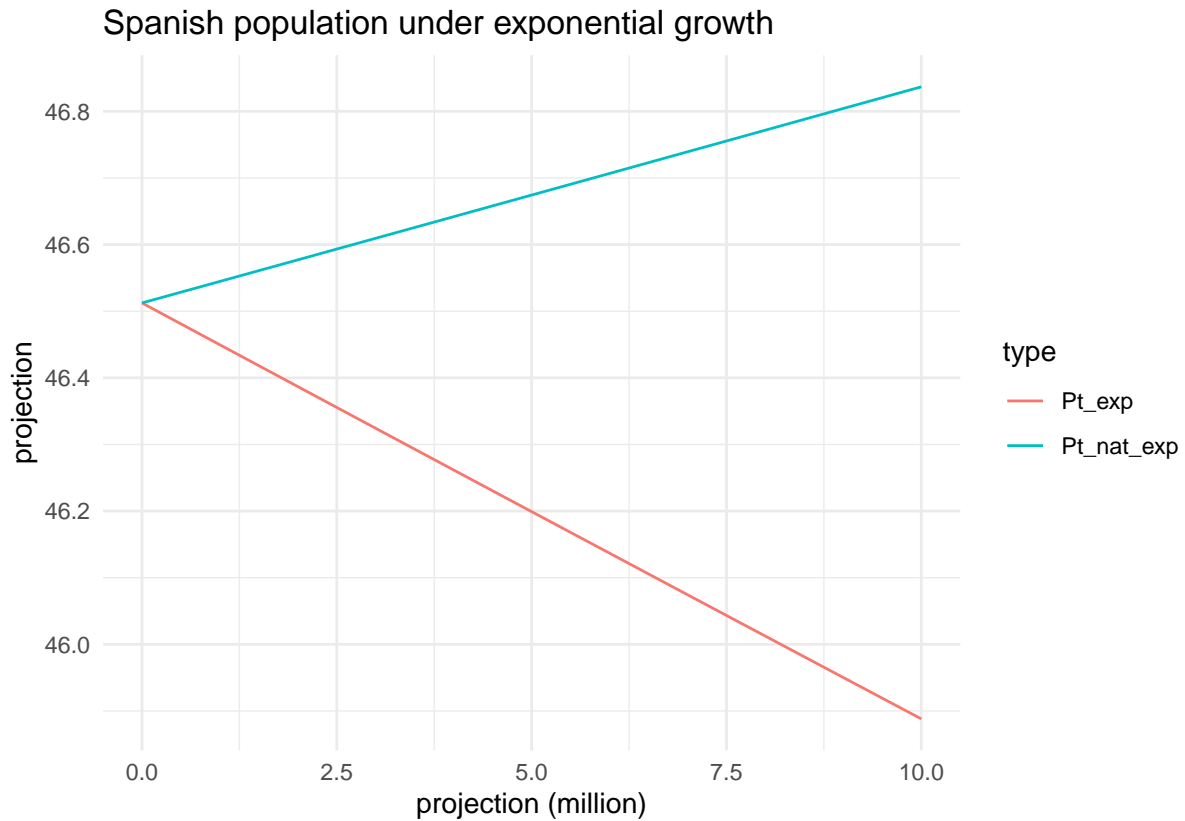
```r
  pivot_longer(-t, names_to = "type", values_to = "projection") %>%
  mutate(projection = projection / 1e6) %>%
  ggplot(aes(x = t, y = projection, color = type)) +
  geom_line() +
  xlab("projection (million)") +
    labs(title = "Spanish population under exponential growth") +
  theme_minimal()
```

Spanish population under exponential growth

3.5 Doubling time

The doubling time is the number of years it would take for a population to double in size, under a fixed rate of change. In an exponential growth situation, if a population double between time 0 and T, then

$$ln(\frac{N(T)}{N(0)}) = ln(2)$$

The doubling time (DB) for a population experiencing an exponential growth at rate r can then be found as:

$$DB = \frac{ln(2)}{r}$$

```r
# Doubling time
exponential_growth %>%
  filter(t == 0) %>%
  mutate(DB = log(2) / r_exp,
```

```
        DB_nat = log(2) / r_nat_exp,
        .keep = "used")
```

```
## # A tibble: 1 x 4
##      r_exp r_nat_exp     DB DB_nat
##      <dbl>     <dbl> <dbl>  <dbl>
## 1 -0.00135  0.000695 -513.   997.
```

4 Fertility and replacement

4.1 Replacement level

The so-called replacement level of fertility is often set at a TFR of 2.1. This is the average number of children a woman would need to have to reproduce herself by bearing a daughter who survives to childbearing age. Roughly, by this logic:

- TFR> 2.1: Population growth
- TFR= 2.1: Replacement level, constant population
- TFR< 2.1: Population decline

As the sex ratio at birth is generally 105 boys per 100 girls and because there is a risk of mortality before reproduction, the TFR has to be greater than 2 for a woman to be able to *replace* herself by a daughter.

- Note: the value 2.1 is a rule of thumb, since it's based on survival it's a moving variable, and also the secondary sex ratio at birth can vary. The lower infant and child mortality is, the closer this value approaches 2. This threshold value can be calculated for a given population, typically by (as usual) abusing the period synthetic population assumption, see e.g. Espenshade, Guzman, and Westoff (2003)

4.2 Gross reproductive rate

The gross reproductive rate (GRR) is similar to the TFR but uses age-specific rate of having a *female* birth ($_nF_x^W[0,T]$)

$$_nF_x^W[0,T] = \frac{Number\ of\ female\ births\ between\ times\ 0\ and\ T\ to\ women\ aged\ x\ to\ x+n}{Number\ of\ person-years\ lived\ by\ women\ aged\ x\ to\ x+n\ between\ times\ 0\ and\ T}$$

$$_nF_x^W[0,T] \approx 0.488 * {_nF_x}$$

$$GRR[0,T] = n \sum_{x=a}^{B-n} {_nF_x^W}[0,T] \approx 0.488 * n \sum_{x=a}^{B-n} {_nF_x}[0,T]$$

```
B <- B2014 |>
  select(age, births)

Fem <-
  ES2014 |>
  filter(sex == "Female")  |>
  select(-births)  |>
  right_join(B, by = "age")  |>
  mutate(exposure = (P2014 + P2015) / 2,
         asfr = births / exposure,
         asfr_f = asfr * .4886)

Fem %>%
  summarize(TFR = sum(asfr),
            GRR = sum(asfr_f))
```

```
## # A tibble: 1 x 2
##     TFR   GRR
##   <dbl> <dbl>
## 1  1.32 0.645
```

4.3 Net reproductive rate

The net reproductive rate (NRR) takes mortality into account. The NRR represents the average
number of daughters that female in a cohort would bear in their lifespan if they were subject
to the observed age-specific fertility ($_nF_x^W$) and mortality rates ($_nM_x$) (Preston, Heuveline, and
Guillot 2001). The measure introduces the person-year lived between age x and x+n ($_nL_x$) in
the GRR measure.

$$NRR[0,T] = n \sum_{x=a}^{B-n} {}_nF_x^W[0,T] * \frac{_nL_x^W}{l_0}$$

```
NRR <-
  Fem %>%
  left_join(LT, by = "age") %>%
  select(age, asfr_f, Lx) %>%
  mutate(Lx = Lx / 1e5) %>%
  summarize(NRR = sum(Lx * asfr_f)) %>%
  pull(NRR)
NRR
```

```
## [1] 0.6402353
```

The NRR is a rate-based measure of growth. NRR > 1 means that the fertility and mortality
would imply growth if held frozen, whereas < 1 means long-term decrease all else equal.

5 Population models

5.1 Stationary vs Stable population

There are two main population growth models in demography: Stable and stationary. These models are theoretical types of population used to understand population dynamics under certain mortality and fertility assumptions.

5.1.1 Stationary population

- Constant number of births
- Constant age-specific death rates
- Zero age-specific net-migration rates

Some properties

- Constant population size and structure
- CGR= 0
- $e_0 = \frac{1}{CBR} = \frac{1}{CDR}$
- All relationships are time-reversible with an alternative interpretation.
- The population structure is proportional to lifetable survivorship

5.1.2 Stationary population alternative definition

- Constant population size and structure
- Constant rates of fertility, mortality, and migration

5.1.3 Stable population

- Constant annual growth in birth: $B(t) = B(0)e^{rt}$
- Constant age-specific death rates
- Zero age-specific net-migration rates

The conditions are then:

- Constant age-specific fertility rates

- Constant age-specific death rates

- Age-specific net-migration rates are zero

- Age-specific net-migration rates are constant

Stationary populations are a special case of stable populations. Stable populations can grow or shrink.

5.2 Leslie matrix

The Leslie matrix is a matrix model used for population projections, assuming a stable population. The model combined mortality and fertility measures you have seen so far. It is a way to assess the long-term consequences, for the size of a population, of keeping a defined set of age-specific fertility and mortality rates constant over time.

It takes the form of

$$
\begin{bmatrix}
N(0,t+1) \\
N(1,t+1) \\
N(2,t+1) \\
N(3,t+1) \\
N(4,t+1) \\
N(5,t+1)
\end{bmatrix}
=
\begin{bmatrix}
F^*(0) & F^*(1) & F^*(2) & F^*(3) & F^*(4) & F^*(5) \\
{}_1S_0 & 0 & 0 & 0 & 0 & 0 \\
0 & {}_1S_1 & 0 & 0 & 0 & 0 \\
0 & 0 & {}_1S_2 & 0 & 0 & 0 \\
0 & 0 & 0 & {}_1S_3 & 0 & 0 \\
0 & 0 & 0 & 0 & {}_1S_4 & {}_+S_5
\end{bmatrix}
*
\begin{bmatrix}
N(0,t) \\
N(1,t) \\
N(2,t) \\
N(3,t) \\
N(4,t) \\
N(5,t)
\end{bmatrix}
$$

where F* are adjusted fertility rates for mortality (see below) and s are the survivorship ratio from age x to x+n.

The Leslie matrix is, generally, conceive to forecast a close (no migration) female-only population.

```
#First, create a 0 matrix (111x111)

nr   <- nrow(LT)
ages <- LT$age
mat  <- matrix(0, nr, nr,
            dimnames = list(to = ages,
                            from = ages))
```

5.2.1 The cohort component method

The Leslie matrix is a matrix representation of the cohort component method. The matrix (see above) is filled in two steps.

Step 1: Project forward the women surviving age-category [x:x+n]

This step consists in finding the survivorship ratio (s) to each age group. The survivorship ratio is the proportion of people age x-n to x year that will be alive n years later, in a stationary population.

$$
s_x = \frac{{}_nL_x}{{}_nL_{x-n}}
$$

```
#Calculate survivorship ratio
Sx <-
  LT |>
  mutate(Sx = lead(Lx) / Lx) |>
  drop_na() |>
  pull(Sx)
```

The survivor at t+n time will thus be the people surviving from age x at time t to age x+n:

$$_nN_x^W(t+n) =_n N_{x-n}^W(t) * s_x$$

To represent this dynamic in the matrix, place the survivorship ratio as a diagonal in the matrix.

```
#Place in matrix as diagonal

mat[row(mat) == col(mat) + 1] <- Sx
```

For the open age interval, the population at time $t + n$ is equal to

$$_\infty N_x^W(t+n) =_n N_{x-n}^W(t) * s_x +_\infty N_x^W(t) * \frac{T_{x+n}}{T_x}$$

```
#The last cell
mat[nr, nr] <-(LT$Tx[nr] / LT$Tx[nr - 1])
```

- It's also common to put a zero in that corner.

Step 2: Finding the number of surviving female in the first age group (age 0)

This step consists in finding the number of female births during the projected period, which survive until the end of the period.

$$_nN_0^W(t+n) = B^W[t,t+n] * \frac{_nL_0}{n * l_0}$$

The way to calculate the number of births is:

$$B[t,t+n] = \sum_{x=\alpha}^{\beta-n} {_nF_x} * n * \frac{(_nN_x^W(t) +_n N_x^W(t+n))}{2}$$

$$= \sum_{x=\alpha}^{\beta-n} {_nF_x} * n * \frac{(_nN_x^W(t) +_n N_{x-n}^W(t)\frac{_nL_x}{_nL_{x-n}})}{2}$$

Thus, $_nN_0^W(t+n)$ is equal to:

$$_nN_0^W(t+n) = \frac{_nL_0}{2 * l_0} \frac{1}{1 + SRB} \sum_{x=\alpha}^{\beta-n} {_nF_x} * n * (_nN_x^W(t) +_n N_{x-n}^W(t)\frac{_nL_x}{_nL_{x-n}})$$

where SRB is the sex ratio at birth.

```r
# get ASFR as vector
ASFR <-
  ES2014 %>%
  filter(sex == "Female") %>%
  select(-births) %>%
  # this is the only difference from before
  # in earlier chunk this was a right_join()
  left_join(B, by = "age") %>%
  mutate(exposure = (P2014 + P2015) / 2,
         asfr = births / exposure,
         asfr = ifelse(is.na(asfr), 0 , asfr)) %>%
  pull(asfr)

# SRB and mortality discount constant
# across age of mother in N(0) equation
PF    <- .4886 # or 0.4878049  = (1 / (1 + 1.05)
const <- PF * (LT$Lx[2] / (2 * LT$lx[1]))

# Non-constant across mother age
firstrow <- const * (ASFR + (lead(ASFR, default = 0) * c(Sx, 0)))

# Fill the first row
mat[1, ] <- firstrow
```

5.2.2 Projections

To forecast the population at time t (P(t)), for 1-year time interval, then

$$P(t) = L^t * P(0)$$

where L is the Leslie matrix and ^ is a matrix power, NOT the kind of power that we're used to! We need %^% from the `expm` package to do this right.

This can work inside a *tidy* framework just as we've done for everything so far in this block, and in this way avoid a bunch of loop-writing and indexing overhead.

```r
# the projection
library(expm)
Pt <-
  ES2014 |>
  filter(sex == "Female") |>
  select(age, Population = P2015) |>
  cross_join(tibble(t = 0:200)) |>
  arrange(t, age) |>
  group_by(t) |>
  mutate(Population = (mat %^% t[1]) %*% Population %>% c()) |>
  ungroup()
```

5.2.3 visualize the results

```r
library(scales) # install if needed
#Total population
Pt %>%
  group_by(t) %>%
  summarize(Population = sum(Population),
            .groups = "drop") %>%
  mutate(Year = 2015 + t) %>%
  ggplot(aes(x = Year, y = Population)) +
  geom_line() +
  scale_y_continuous(labels = label_number(suffix = " M", scale = 1/1000000)) +
  labs(title = "Population size for Spanish female projection")
```
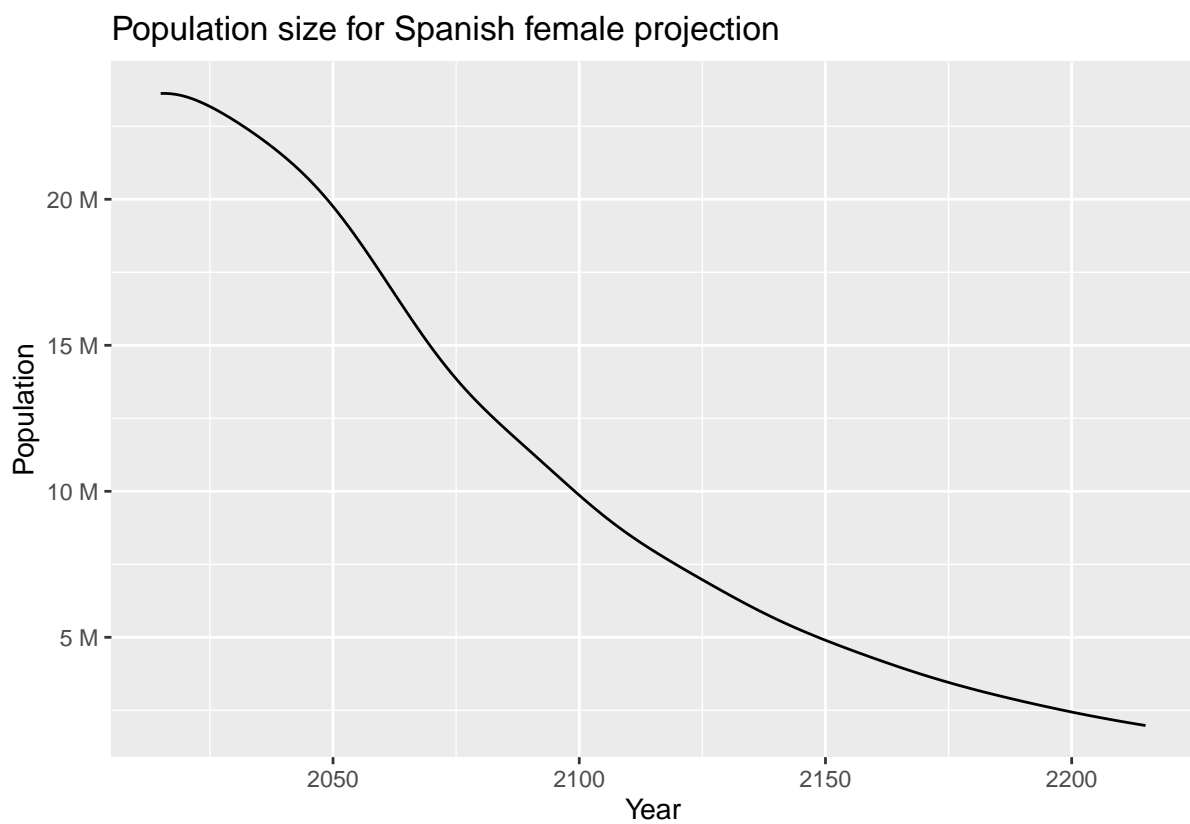


```r
#Number of births
Pt %>%
  filter(age == 0) |>
  mutate(year = 2015 + t) |>
  ggplot(aes(x = year, y = Population)) +
  geom_line() +
  scale_y_continuous(labels = label_number(suffix = " K", scale = 1/1000)) +
  labs(title = "Number of births in Spanish female projection")
```

Number of births in Spanish female projection
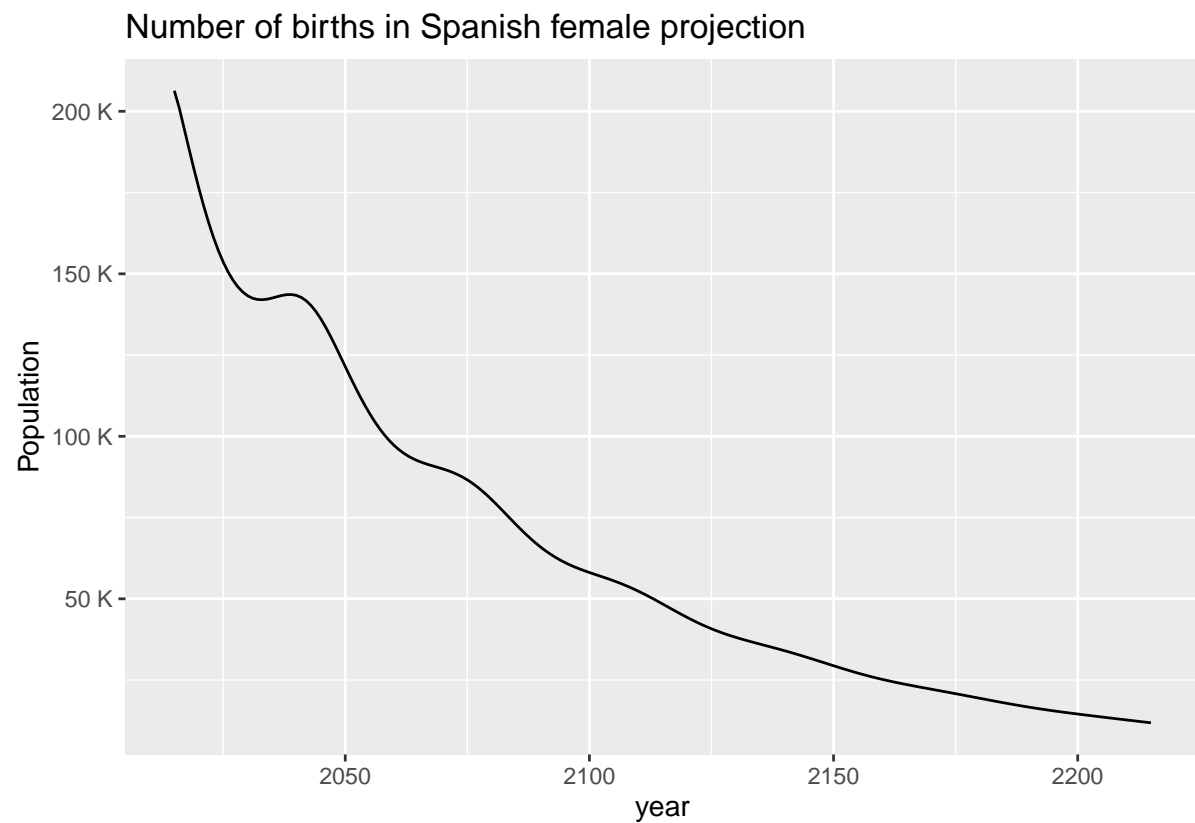

```r
# structure in a given year

Pt %>%
  filter(t == 10)|>
  ggplot() +
  geom_bar(aes(x = age,
               y = Population),
           stat = "identity",
           fill = "#e32951",
           width = 1) +
  ylab("Population counts")+
  coord_flip()+
  theme_bw() +
  scale_y_continuous(labels = label_number(suffix = " K", scale = 1/1000)) +
  labs(title = "Spanish female projection in year 2025")
```

Spanish female projection in year 2025
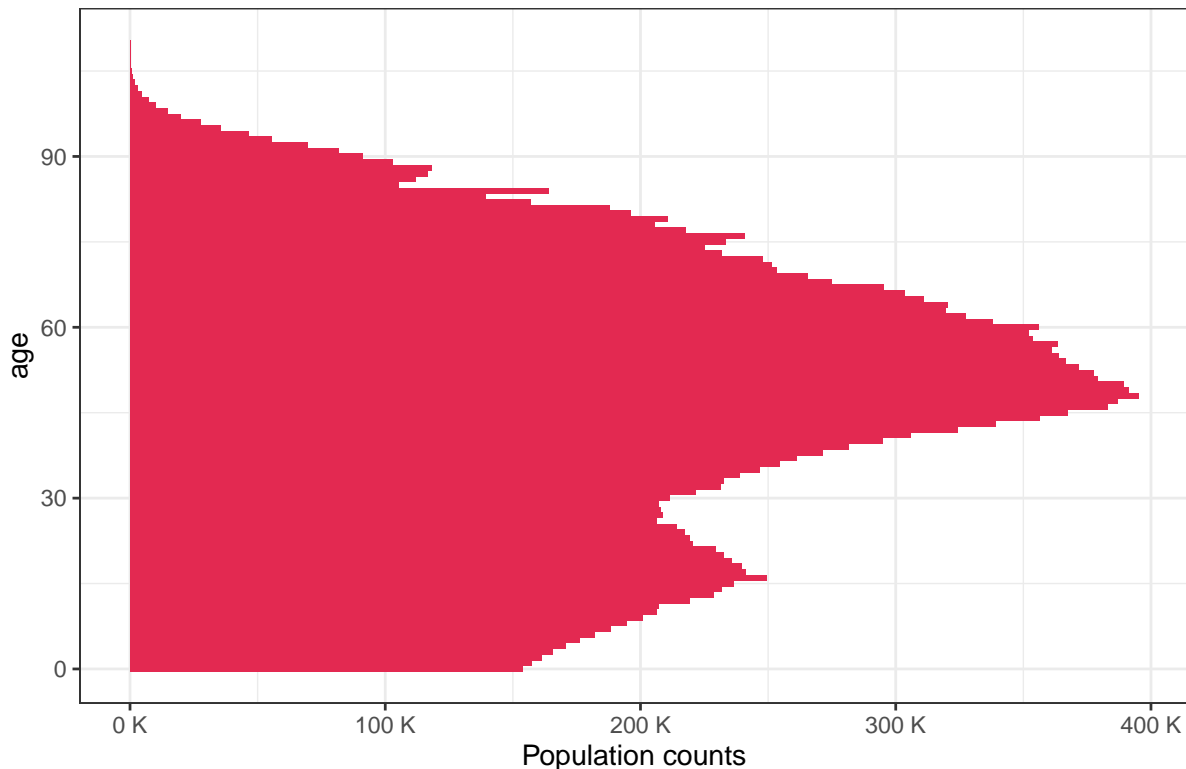
5.3 Leslie matrix and stable population properties

5.3.1 Constant population structure and crude growth rate

In the long term, the population age structure converges towards a constant age-structure.

```
library(colorspace)
Struct_t <-
  Pt %>%
  filter(t %% 20 == 0) %>%
  mutate(Year = 2015 + t) %>%
  group_by(t) %>%
  mutate(Structure = 100 * Population / sum(Population)) %>%
  ungroup()

Struct_t %>%
  ggplot(aes(x = age, y = Structure, color = Year, group = Year)) +
  geom_line(size = 2, alpha = .8) +
  scale_color_continuous_sequential("BurgYl", begin = .2, end = 1)
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```
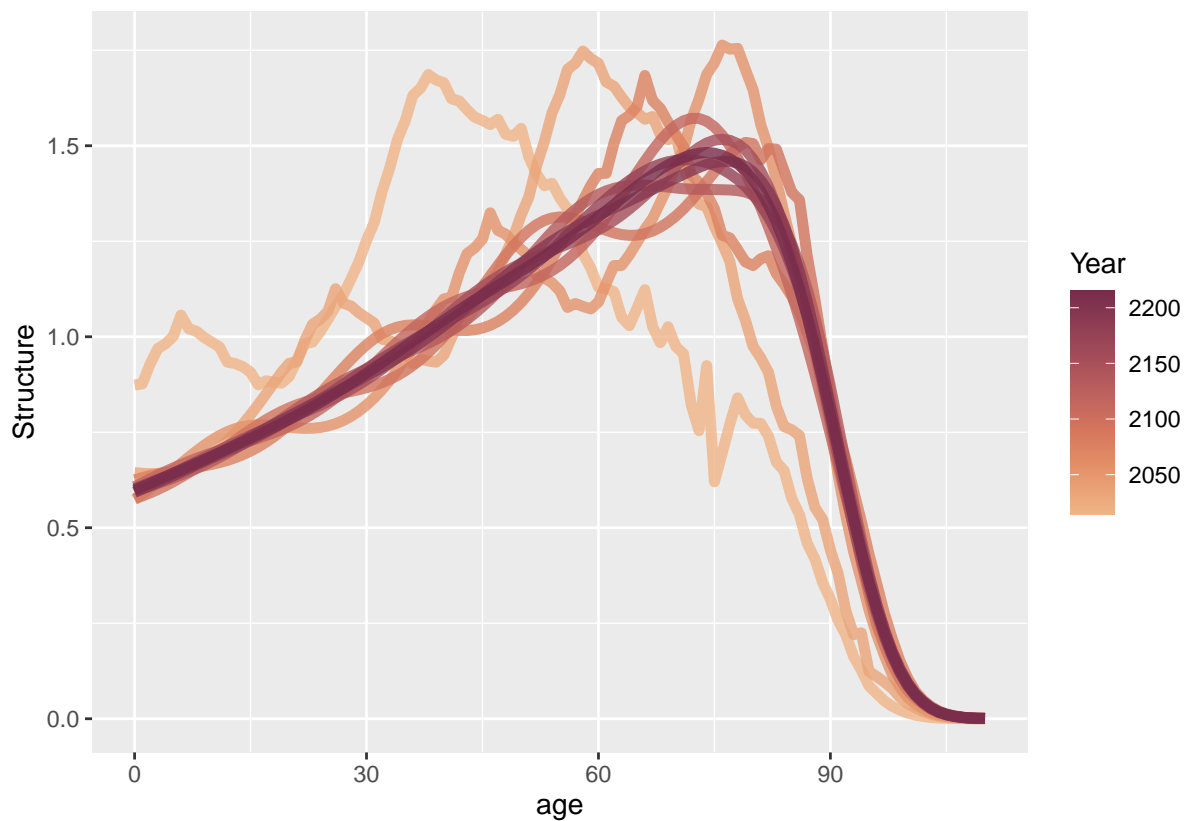
The crude growth rate also converges towards a constant over time in a stable population.

```
CGRt <-
  Pt %>%
  group_by(t) %>%
  summarize(Population = sum(Population),
            .groups = "drop") %>%
  drop_na() %>%
  mutate(CGR = log(lead(Population) / Population))

CGRt %>%
  ggplot(aes(x = t, y = CGR)) +
  geom_line() +
  labs(title = "Projected crude growth rate of Spanish female population",
       subtitle = "From 2014 natural growth components") +
  geom_hline(yintercept = log(NRR) / 33, color = "red") +
  annotate("text",
           y = -0.0115,
           x = 100,
           label = "Tim's back of envelope guess from class",
           color = "red")
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## ('geom_line()').
```

Projected crude growth rate of Spanish female population
From 2014 natural growth components
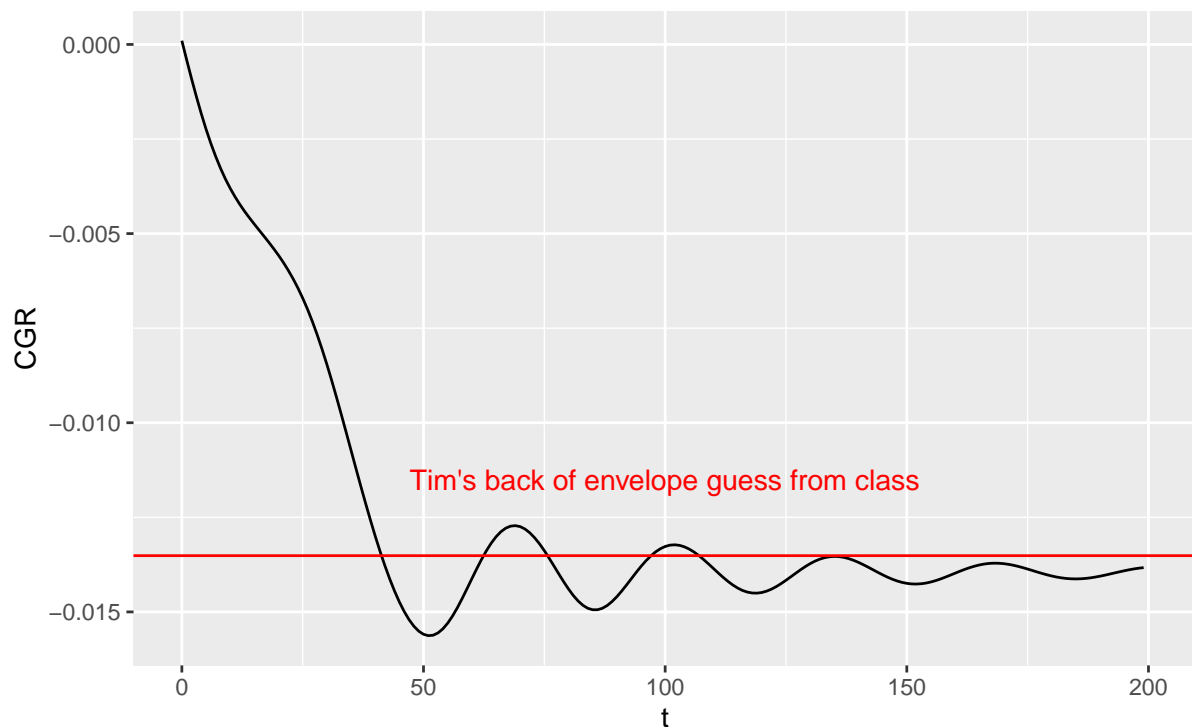
Eigenvalue as crude growth rate in stable populations

Eigen comes from the German "self". The eigenvalues can be referred to as "latent roots" of a squared matrix. They play a fundamental role in the solution of linear system and provide concrete information "*from a set of static, algebric equations*" (Caswell 2001).

The eigenvalue (λ) of a matrix A is that

$$Ax = \lambda x$$

where x is some scalar, called eigenvector.

The eigenvalue is found by solving

$$(A - \lambda I)x = 0$$

where I is the identity matrix.

For example (coming from Caswell (2001))

```
#Example
A<-matrix(c(3,2,-6,-5), ncol=2)
A
```

```
##      [,1] [,2]
## [1,]    3   -6
## [2,]    2   -5
```

18

```r
#if x=t(1,1)
x<-matrix(c(1,1), ncol=1)
x
```

```
##      [,1]
## [1,]    1
## [2,]    1
```

```r
#then lambda
lambda<- A%*%x

#x is thus the eigenvector of A and lambda is -3
lambda
```

```
##      [,1]
## [1,]   -3
## [2,]   -3
```

The intrinsic crude growth rate of a stable population is found by finding the eigenvalue of the Leslie matrix, $CGR = 1 - \lambda$.

```r
# asymptotic growth rate
r <- -(1-as.numeric(eigen(mat)$values[1]))
r
```

```
## [1] -0.01385219
```

```r
# crude growth after 200 years
CGRt %>%
  slice(n()-1) %>%
  pull(CGR)
```

```
## [1] -0.01382947
```

```r
# go another 100 years and it will get even closer!

# back of envelope (change 33 to get closer)
log(NRR) / 33
```

```
## [1] -0.01351271
```

```r
# what would have been the perfect guess?
log(NRR) / r
```

```
## [1] 32.19126
```

The *intrinsic* population structure is the first eigenvector of the Leslie matrix. If you rescale this, then you can compare it to the projected population structure. If you project far enough into the future there will be perfect overlap.

```
evec          <- eigen(mat)$vector[, 1]  |>  Re()
stable_struct <- tibble(age = 0:110,
                        Stable = 100 * evec / sum(evec))
Struct_t %>%
  filter(t == max(t)) |>
  left_join(stable_struct, by = "age") |>
  pivot_longer(Structure:Stable,
               names_to = "variant",
               values_to = "Structure") |>
  ggplot(aes(x = age, y = Structure, color = variant, group = variant)) +
  geom_line()
```
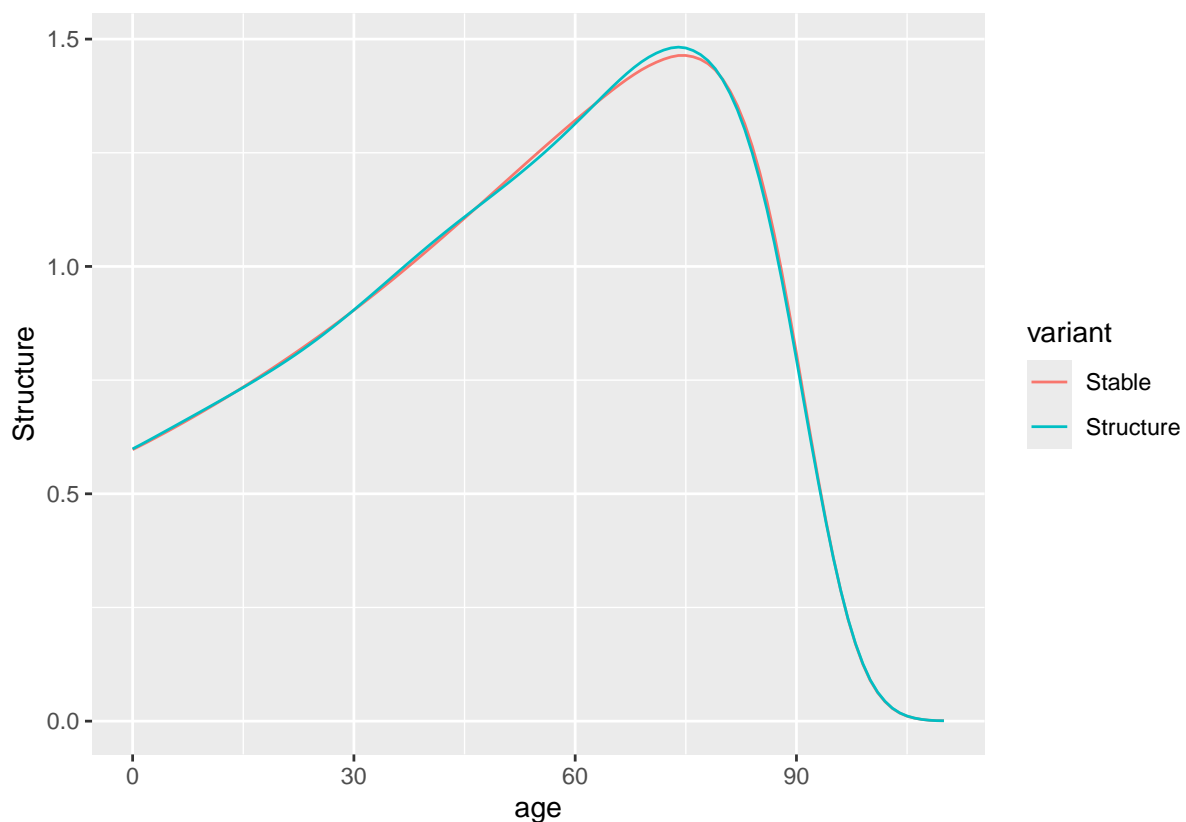

Exercises

1) Load 'LTFrance2015.txt', 'ASFRFrance2015.txt' and 'PopFrance2015.txt'.

2) Calculate the TFR and NRR for France in 2015. Discuss if France reached the population replacement level in 2015.

3) Create the Leslie matrix for French females in 2015.

4) Project French females population 200 years ahead. What is the proportion of the 2015 population left 200 later?

5) How did the age-structure change between 2015 and 2215? Older or younger?

6) Calculate the CGR for France. Discuss your results in relation with the CGR for Spain found in the class.

References

Caswell, Hal. 2001. *Matrix Population Models*. Wiley Online Library.

Espenshade, Thomas J, Juan Carlos Guzman, and Charles F Westoff. 2003. "The Surprising Global Variation in Replacement Fertility." *Population Research and Policy Review* 22: 575–83.

Human Fertility Database. 2018. "Max Planck Institute for Demographic Research (Germany) and Vienna Institute of Demography (Austria)."

Human Mortality Database. 2018. "University of California, Berkeley (USA) and Max Planck Institute for Demographic Research (Germany)."

Preston, S, Patrick Heuveline, and Michael Guillot. 2001. "Demography: Measuring and Modeling Population Processes." *Malden, MA: Blackwell Publishers.*