

Introducción al Paquete Demography

Eduardo Bologna

21/1/2020

<https://www.rdocumentation.org/packages/demography/versions/1.22/topics/demogdata>
<https://rpubs.com/Timexpo/487053>

Para esta sección, se recomienda crear un nuevo proyecto, específico para las operaciones que se realizarán con este paquete. De ese modo se pueden tener los datos fácilmente accesibles.

Descripción del paquete: “El paquete R **demography** provee funciones para el análisis demográfico, que incluyen: elaboración de tablas de mortalidad, modelización de Lee-Carter, análisis funcional de datos sobre tasas de mortalidad, fecundidad y migración neta, y proyecciones estocásticas de población”

Elaboración de tablas de mortalidad: una tabla de mortalidad describe la demografía de una población en términos de supervivencia, esencialmente, el número de individuos que se espera que alcancen a la próxima edad o la próxima etapa de su vida.

Modelización de Lee-Carter: Un enfoque para proyectar mortalidad, que no es frecuente en los paquetes estadísticos.

Análisis funcional de datos: Permite el análisis de diferentes indicadores de una población: tasas de mortalidad y fecundidad, migración neta y proyecciones estocásticas de población.

Primer ejemplo

La instalación del paquete se realiza por el medio habitual

```
install.packages("demography")
```

Y se carga en la sesión

```
library(demography)
```

```
## Warning: package 'demography' was built under R version 3.5.3
```

```
## Loading required package: forecast
```

```
## Warning: package 'forecast' was built under R version 3.5.3
```

```
## This is demography 1.22
```

Para importar las bases de datos a R se usa el comando `read.demogdata()` del paquete `demography`. La estructura del comando es:

```
read.demogdata(archivo de tasas, archivo de población, type, label, max.mx = 10, skip = 2, popskip = (p
```

El *archivo de tasas* es el archivo que contiene las tasas que interesan: mortalidad o fecundidad El *archivo de población* contiene la composición de la población por sexo y edad *Type* es el componente de la dinámica demográfica que se analiza, puede valer: “mortality”, “fertility” o “migration”. *Label* es el nombre del área que se analiza *Max.mx* establece un límite superior para las tasas bajo análisis; su función es la de tratar a todo valor que sea mayor que ese número, como igual a él. *Skip* indica cuántas líneas deben saltarse antes de empezar a leer el archivo, para el caso de archivos que tengan títulos antes de los nombres de las columnas. *Popskip* es lo mismo para el archivo de población. *Lambda* es un valor que se usa como parámetro en la transformación de Box-Cox Es un modo de convertir datos no normales en normales. Los valores por defecto son 0 para mortalidad, 0.4 para fecundidad y 1 para migración *Scale* indica la escala de los datos, si las tasas son por individuo “scale=1” si son por mil “scale=1000”.

Vamos a usar datos del proyecto The Human Mortality Database For this tutorial, data will be obtained from the human mortality database (Shkolnikov, 2020). <https://www.mortality.org/> Una vez hecho el registro (sin costo) se pueden descargar los datos para 41 países o áreas y diversos períodos. Para este ejemplo se usarán los datos de USA (UNA VEZ RECORRIDO, SE CAMBIA POR LA DE CHILE). Las bases que nos van a interesar por el momento son las de tasas de mortalidad y exposición al riesgo, ambas en la sección “datos periódicos”. Para aprovechar las funciones de este paquete, es necesario adecuar otras tablas a este formato.

Una vez que se han descargado las dos bases (tasas y población sometida al riesgo) en la carpeta donde se encuentra el proyecto, conviene observar los archivos .txt, para tener en mente la estructura general de los datos.

Luego las bases son leídas convirtiéndolas en un objeto de clase “demogdata”, que será usado por varias funciones del paquete.

Hemos usado los nombres con que vienen por defecto las bases de HMD: Mx_1x1.txt para las tasas de mortalidad y Exposures_1x1.txt para la población expuesta.

```
USA_mortalidad<-read.demogdata(  
  file="Mx_1x1.txt",popfile="Exposures_1x1.txt", type="mortality",  
  label="USA",skip=2, scale=1 )
```

Estos datos se pueden visualizar con el comando plot, con estructura

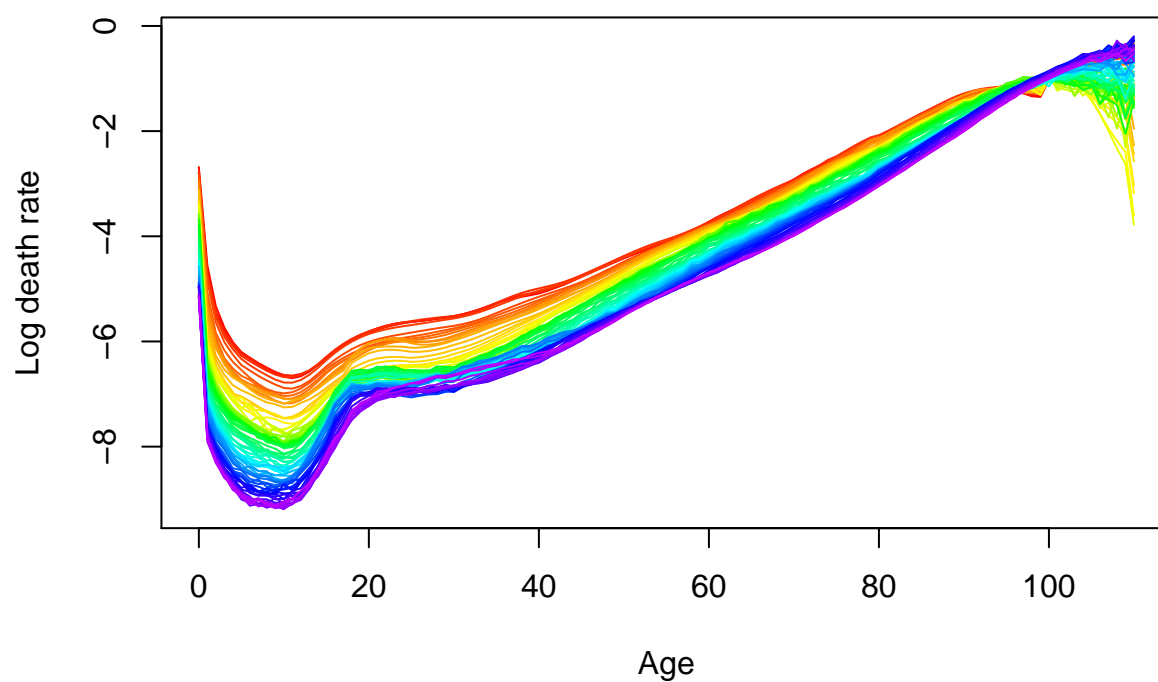
```
plot(x, series, datatype, years, ages, max.age,  
  transform, plot.type, type = "l", main, xlab, ylab,)
```

cuyos argumentos son: *x*: el objeto de clase demogdata con que se está trabajando. *series*: el nombre de la serie que se va a graficar. Si no se especifica, por defecto toma la primera matriz, según *datatype*. *datatype*: por defecto es tasas (“rate”), la opción es especificar población *datatype*=“pop”. *years*: un vector que especifica los años a graficar, por defecto, toma todos los que están en la serie *ages*: igual que los años, si no se indica, se toman todas las edades disponibles *max.age*: la máxima edad a graficar, por defecto, las disponibles *transform*: lógico, por defecto TRUE, que indica que los datos se transforman antes de graficar *plot.type*: tipo de gráfico pueden ser funciones o tiempo *main*: título del gráfico *xlab* e *ylab*: rótulos de los ejes

La mayoría de estos argumentos tienen sus valores por defecto y, para un gráfico básico, alcanza con indicar el origen de los datos y la serie elegida. Así, para el logaritmo de tasas de mortalidad (en el eje vertical) por edades (eje horizontal) y año (representado en el color), se pide simplemente:

```
plot(USA_mortalidad, series= "Total")
```

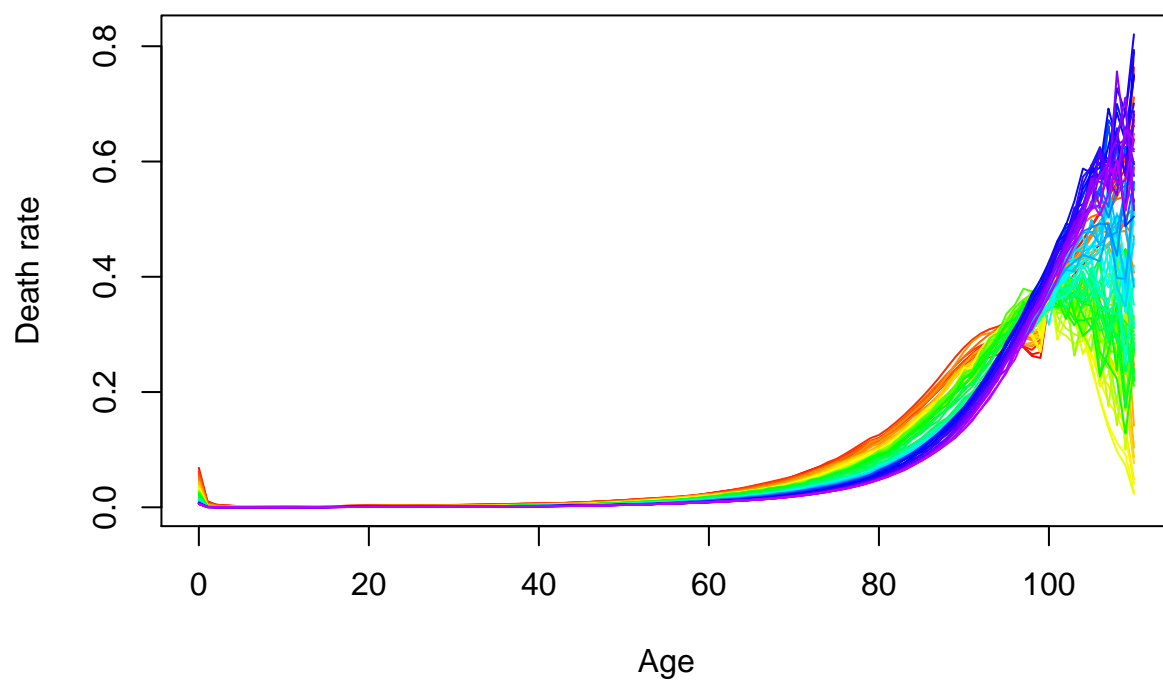
USA: total death rates (1933–2017)



Si no se quiere la transformación logarítmica, se indica:

```
plot(USA_mortalidad, series= "Total", transform = FALSE)
```

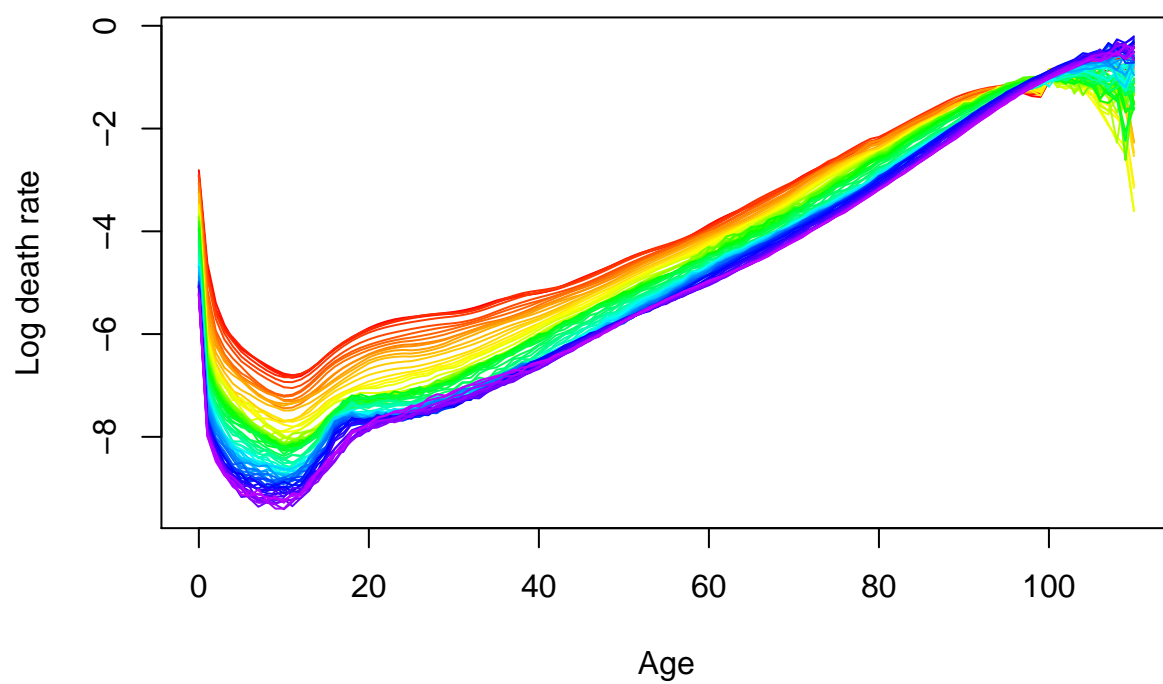
USA: total death rates (1933–2017)



Dado que la transformación logarítmica permite apreciar mejor las diferencias y, si se solicita solo de mujeres:

```
plot(USA_mortalidad, series= "Female")
```

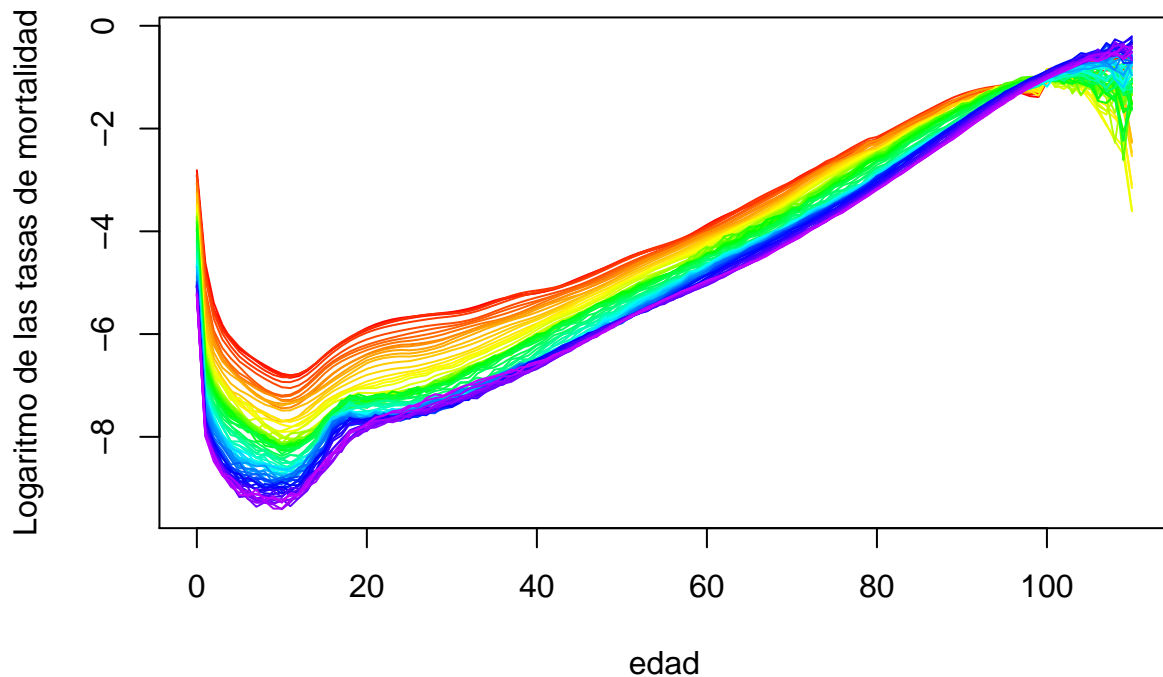
USA: female death rates (1933–2017)



Se pueden ajustar las opciones de formato por defecto

```
plot(USA_mortalidad, series= "Female", main = "Tasas de mortalidad femeninas en Estados Unidos (1933-2017)",
```

Tasas de mortalidad femeninas en Estados Unidos (1933–2017)

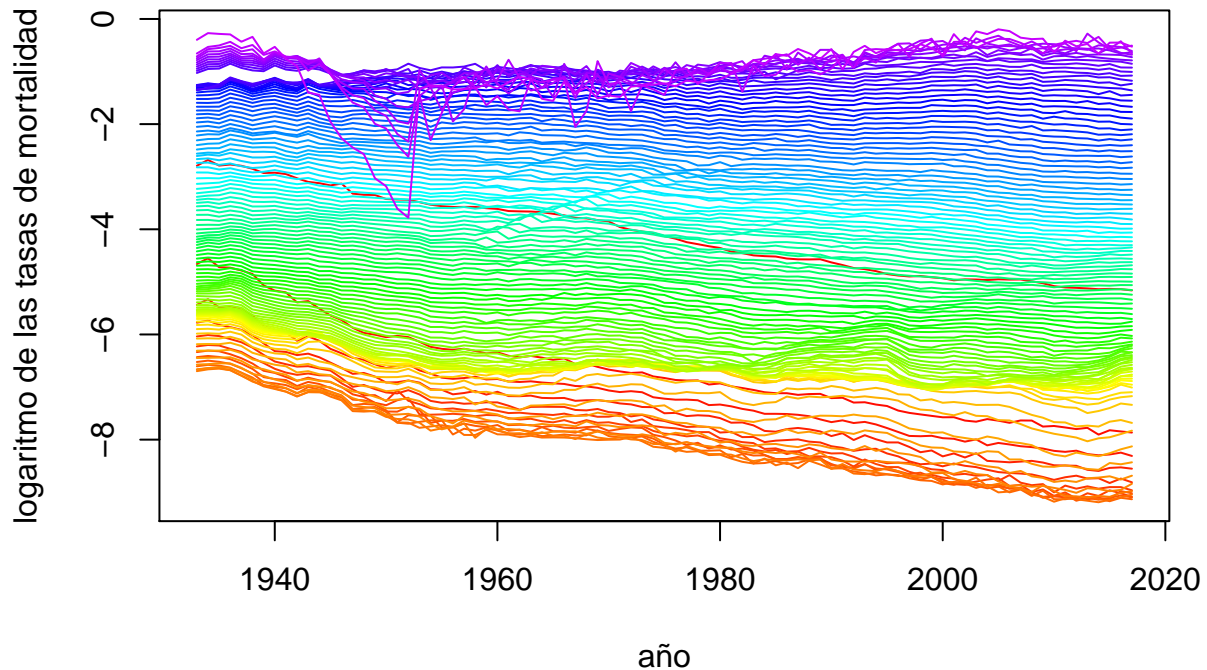


En todos los gráficos se ve la tendencia general según la cual las tasas tienen un comportamiento similar a lo largo de la vida de las personas y decrecen a lo largo de la historia. El apartamiento del patrón por edades se observa en las edades mayores, donde la caída de la mortalidad aparece más acentuada.

Si se cambia el tipo de gráfico por “tiempo”, se visualiza cómo se modifican las tasas específicas de mortalidad por edad a lo largo del tiempo. Ahora cada color representa una edad y el eje horizontal es la línea de tiempo. Ajustamos los nombres de los ejes:

```
plot(USA_mortalidad, series= "Total", plot.type = "time",  
     main="USA: Evolución de las tasas específicas \n de mortalidad por edad (1933-2017)", xlab = "año")
```

USA: Evolución de las tasas específicas de mortalidad por edad (1933–2017)



Al que le haría falta un pequeño ajuste de formato, porque el paquete mantiene el nombre del eje. Se puede modificar con `xlab`.

Luego de haberse registrado en HMD, hay un comando R que permite acceder a los datos directamente desde la página. El comando es:

```
hmd.mx(country,username,password, label=country)
```

En el que: *country* es el nombre abreviado del país que se solicita *username* y *password*, son los que corresponden a la cuenta creada *label* es la etiqueta, que debe coincidir con el nombre del país que se pidió. (todo va entre comillas)

Tablas de mortalidad

El comando para producir una tabla de mortalidad es `lifetable`, con estructura:

```
lifetable(x, series = names(data$rate)[1], years = data$year,
  ages = data$age, max.age = min(100, max(data$age)),
  type = c("period", "cohort"))
```

`data` refers to a demogdata file `series` refers to the name of the series to use, defaults to one. In this instance 1 would be female, 2 would be male, 3 would be total years=vector of which years to include ages=vector of which ages to include max.age=age for last row, subsequent rows are combined type=period or cohort data A period lifetable examines all age groups at the current time A cohort lifetable examines a cohort over a long period of time

Para construir la tabla de mortalidad para USA en 2016 y guardarla con el nombre “tabla_de_mortalidad”, se solicita

```
tabla_de_mortalidad=lifetable(USA_mortalidad, series="Total", years=2016)
```

El objeto resultante es una lista que contiene las columnas usuales de una tabla de mortalidad:

```
names(tabla_de_mortalidad)
```

```
## [1] "age"      "year"     "mx"       "qx"       "lx"       "dx"       "Lx"       "Tx"
## [9] "ex"       "rx"       "series"   "type"     "label"
```

mx= Age or class specific death rates qx= Probability of death between this age and the age + 1 lx= Proportion of the starting population surviving to the next stage dx= Proportion of the starting populations dying during this stage Lx= The number of person years lived relative to the starting population of that year Tx= Average person-years left to live after this year ex= Expectation of life at age x, very similar to Tx for this dataset If label la etiqueta del país que se analiza

Y se los puede ver impresos individualmente, por ejemplo, la esperanza de vida para cada edad:

```
tabla_de_mortalidad$ex
```

```
##          2016
## 0  78.875744
## 1  78.338066
## 2  77.368585
## 3  76.388727
## 4  75.403755
## 5  74.415740
## 6  73.425793
## 7  72.434618
## 8  71.443682
## 9  70.451912
## 10 69.459677
## 11 68.467401
## 12 67.475694
## 13 66.484334
## 14 65.494694
## 15 64.508345
## 16 63.526206
## 17 62.549719
## 18 61.579944
## 19 60.620698
## 20 59.665866
## 21 58.714283
## 22 57.769905
## 23 56.826229
## 24 55.884556
## 25 54.943572
## 26 54.004583
## 27 53.064328
## 28 52.127022
## 29 51.191803
```


30 50.254344
31 49.318519
32 48.384224
33 47.452220
34 46.519692
35 45.589603
36 44.660178
37 43.732528
38 42.805393
39 41.881017
40 40.956773
41 40.033025
42 39.111338
43 38.192120
44 37.280083
45 36.371300
46 35.466665
47 34.567584
48 33.671478
49 32.784520
50 31.905425
51 31.034042
52 30.169792
53 29.316217
54 28.471609
55 27.636060
56 26.808787
57 25.988138
58 25.177065
59 24.373508
60 23.579779
61 22.791231
62 22.010626
63 21.238332
64 20.472119
65 19.716021
66 18.962889
67 18.213813
68 17.474282
69 16.745474
70 16.026543
71 15.316862
72 14.620426
73 13.942241
74 13.278113
75 12.627699
76 11.990523
77 11.362347
78 10.752823
79 10.157309
80 9.580387
81 9.024818
82 8.486774
83 7.968045

```
## 84 7.465084
## 85 6.986932
## 86 6.532340
## 87 6.098584
## 88 5.684033
## 89 5.286357
## 90 4.901044
## 91 4.541950
## 92 4.212051
## 93 3.905425
## 94 3.625232
## 95 3.367656
## 96 3.124947
## 97 2.931744
## 98 2.732745
## 99 2.572675
## 100 2.417898
```

Si hubiésemos pedido más años, por ejemplo el inicio y fin de la serie disponible:

```
tabla_de_mortalidad_compara=lifetable(USA_mortalidad, series="Total", years=c(1933,2016))
```

Las esperanzas de vida por edades resultan:

```
tabla_de_mortalidad_compara$ex
```

```
##      1933      2016
## 0 60.906901 78.875744
## 1 63.676859 78.338066
## 2 63.277289 77.368585
## 3 62.551028 76.388727
## 4 61.743934 75.403755
## 5 60.890237 74.415740
## 6 60.009625 73.425793
## 7 59.115289 72.434618
## 8 58.209795 71.443682
## 9 57.294633 70.451912
## 10 56.371585 69.459677
## 11 55.442475 68.467401
## 12 54.510316 67.475694
## 13 53.579385 66.484334
## 14 52.653642 65.494694
## 15 51.736319 64.508345
## 16 50.829654 63.526206
## 17 49.933440 62.549719
## 18 49.046822 61.579944
## 19 48.169217 60.620698
## 20 47.300248 59.665866
## 21 46.439358 58.714283
## 22 45.585033 57.769905
## 23 44.735416 56.826229
## 24 43.888603 55.884556
## 25 43.042601 54.943572
```

26 42.195946 54.004583
27 41.348707 53.064328
28 40.501316 52.127022
29 39.654052 51.191803
30 38.807169 50.254344
31 37.961013 49.318519
32 37.116799 48.384224
33 36.275878 47.452220
34 35.439676 46.519692
35 34.609546 45.589603
36 33.785629 44.660178
37 32.966059 43.732528
38 32.148869 42.805393
39 31.332839 41.881017
40 30.517524 40.956773
41 29.703698 40.033025
42 28.893906 39.111338
43 28.090645 38.192120
44 27.295463 37.280083
45 26.509283 36.371300
46 25.732538 35.466665
47 24.964834 34.567584
48 24.205794 33.671478
49 23.455707 32.784520
50 22.715118 31.905425
51 21.984141 31.034042
52 21.261649 30.169792
53 20.546193 29.316217
54 19.836342 28.471609
55 19.130836 27.636060
56 18.429592 26.808787
57 17.735768 25.988138
58 17.053270 25.177065
59 16.385805 24.373508
60 15.736971 23.579779
61 15.108708 22.791231
62 14.497826 22.010626
63 13.900054 21.238332
64 13.311713 20.472119
65 12.729466 19.716021
66 12.151915 18.962889
67 11.583483 18.213813
68 11.030136 17.474282
69 10.497787 16.745474
70 9.992694 16.026543
71 9.519023 15.316862
72 9.072367 14.620426
73 8.645844 13.942241
74 8.233184 13.278113
75 7.826092 12.627699
76 7.414556 11.990523
77 7.016866 11.362347
78 6.637047 10.752823
79 6.279035 10.157309

```
## 80  5.945614  9.580387
## 81  5.626863  9.024818
## 82  5.331649  8.486774
## 83  5.057955  7.968045
## 84  4.804028  7.465084
## 85  4.568504  6.986932
## 86  4.349673  6.532340
## 87  4.150186  6.098584
## 88  3.974772  5.684033
## 89  3.826648  5.286357
## 90  3.706287  4.901044
## 91  3.610975  4.541950
## 92  3.534401  4.212051
## 93  3.469224  3.905425
## 94  3.407633  3.625232
## 95  3.344999  3.367656
## 96  3.285874  3.124947
## 97  3.212170  2.931744
## 98  3.088239  2.732745
## 99  2.870801  2.572675
## 100 2.576604  2.417898
```

Solo para mujeres:

```
tabla_de_mortalidad_compara_mujeres=lifetable(USA_mortalidad, series="Female", years=c(1933,2016))
tabla_de_mortalidad_compara_mujeres$ex
```

```
##           1933           2016
## 0  62.811055  81.364782
## 1  65.239966  80.798508
## 2  64.816506  79.827100
## 3  64.075902  78.846379
## 4  63.258563  77.859385
## 5  62.398671  76.871222
## 6  61.513414  75.881151
## 7  60.613863  74.889066
## 8  59.702416  73.898068
## 9  58.780615  72.905115
## 10 57.850243  71.912790
## 11 56.913249  70.919432
## 12 55.972965  69.927319
## 13 55.033797  68.934887
## 14 54.099878  67.943647
## 15 53.174541  66.955518
## 16 52.260312  65.968412
## 17 51.357039  64.984454
## 18 50.463900  64.003609
## 19 49.580401  63.027490
## 20 48.706208  62.051693
## 21 47.840843  61.078228
## 22 46.982699  60.107555
## 23 46.129957  59.138022
## 24 45.280693  58.169695
```

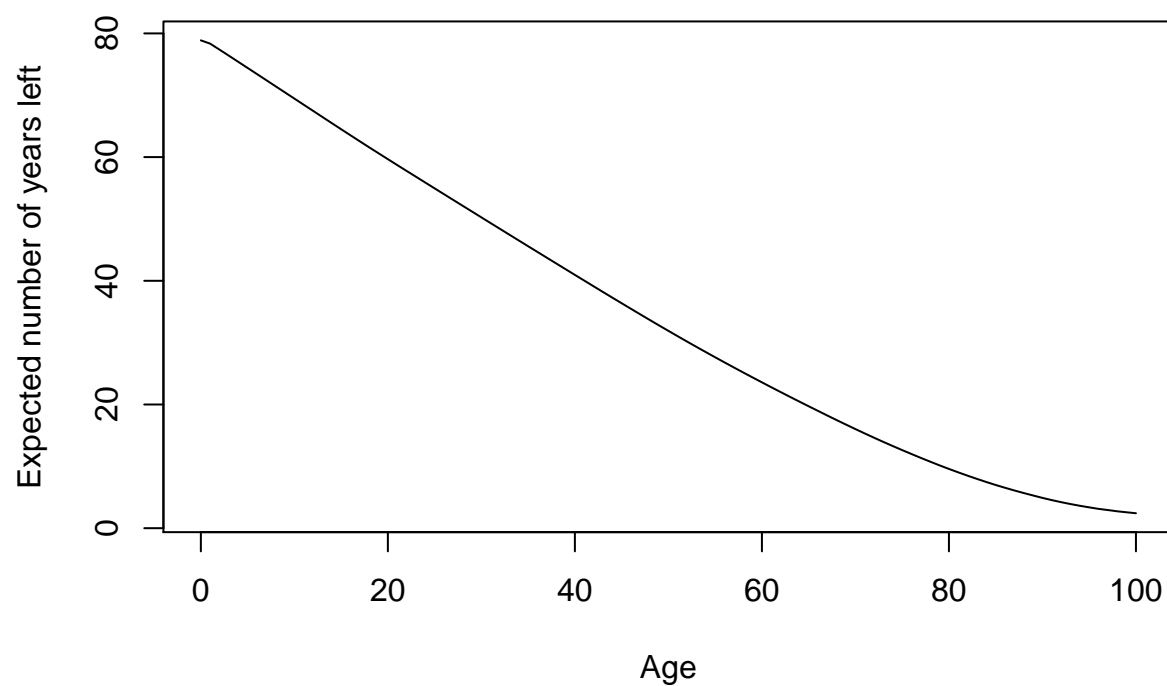
25 44.433027 57.200787
26 43.585519 56.234591
27 42.737966 55.270105
28 41.890467 54.307416
29 41.042857 53.345741
30 40.194893 52.384491
31 39.346618 51.425652
32 38.499118 50.470231
33 37.653861 49.516375
34 36.812362 48.562276
35 35.976064 47.610747
36 35.145226 46.661230
37 34.317841 45.714299
38 33.491739 44.767458
39 32.665468 43.824821
40 31.838241 42.882004
41 31.010699 41.941338
42 30.185383 41.001389
43 29.364957 40.066608
44 28.551070 39.136029
45 27.744869 38.208888
46 26.946845 37.287593
47 26.156710 36.370421
48 25.374119 35.454456
49 24.599450 34.546683
50 23.833405 33.647337
51 23.076256 32.753280
52 22.326996 31.865911
53 21.584294 30.985045
54 20.846730 30.112862
55 20.113029 29.246450
56 19.383025 28.386700
57 18.659709 27.533182
58 17.946759 26.685156
59 17.247671 25.843214
60 16.565805 25.008227
61 15.903125 24.178783
62 15.257184 23.351246
63 14.624656 22.534191
64 14.002724 21.722303
65 13.388807 20.918408
66 12.781997 20.117988
67 12.186119 19.323855
68 11.606208 18.542286
69 11.047248 17.770713
70 10.514513 17.009023
71 10.011520 16.258698
72 9.534744 15.518973
73 9.078685 14.798408
74 8.638430 14.093116
75 8.207093 13.400230
76 7.775680 12.719677
77 7.361473 12.047018
78 6.967974 11.396098

```
## 79    6.598638 10.761938
## 80    6.255613 10.144729
## 81    5.919593  9.551013
## 82    5.606368  8.974904
## 83    5.313547  8.420648
## 84    5.039102  7.881476
## 85    4.781830  7.365294
## 86    4.540413  6.878542
## 87    4.317903  6.409451
## 88    4.119429  5.962340
## 89    3.948101  5.537632
## 90    3.804875  5.127217
## 91    3.689143  4.745027
## 92    3.597437  4.394854
## 93    3.526981  4.067653
## 94    3.476438  3.766602
## 95    3.441207  3.491186
## 96    3.410900  3.231483
## 97    3.359084  3.021537
## 98    3.250376  2.813335
## 99    3.052195  2.638580
## 100   2.781704  2.472343
```

La visualización de estos resultados se logra con el comando `plot`, que, por defecto grafica la esperanza de vida por edad:

```
plot(tabla_de_mortalidad)
```

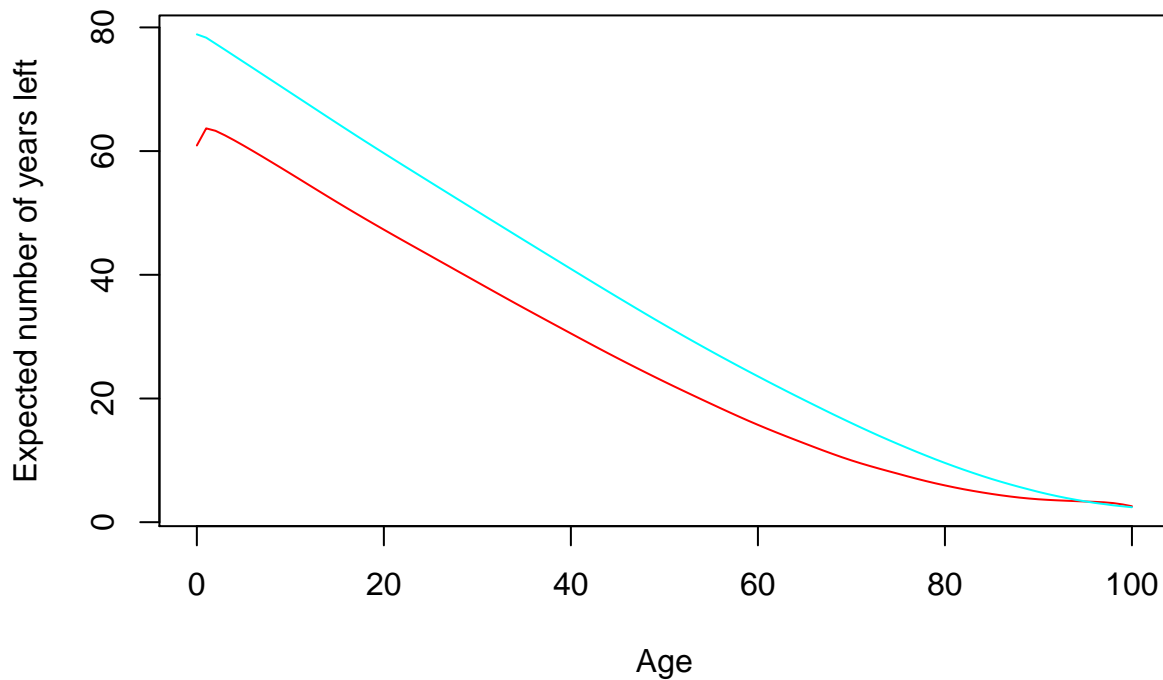
Life expectancy: USA Total (2016)



Al que se pueden ajustar título y nombres de los ejes.
No hay inconveniente en graficar más de un año:

```
plot(tabla_de_mortalidad_compara)
```

Life expectancy: USA Total (1933–2016)



El comando `life.expectancy` provee esperanzas de vida para determinadas edades y años; por defecto toma edad cero años y todos los años disponibles. Por ejemplo:

```
ev_nac<-life.expectancy(USA_mortalidad, series = "Total")
ev_nac
```

```
## Time Series:
## Start = 1933
## End = 2017
## Frequency = 1
##      1933      1934      1935      1936      1937      1938      1939      1940
## 60.90690 60.19977 60.91639 60.36011 61.04395 62.39958 63.07703 63.22324
##      1941      1942      1943      1944      1945      1946      1947      1948
## 63.79659 64.54374 64.36603 65.06795 65.59747 66.13765 66.81310 67.22941
##      1949      1950      1951      1952      1953      1954      1955      1956
## 67.63678 68.06699 68.17809 68.38618 68.72044 69.50739 69.56082 69.64528
##      1957      1958      1959      1960      1961      1962      1963      1964
## 69.41628 69.66965 69.90069 69.82200 70.24846 70.11187 69.93943 70.19788
##      1965      1966      1967      1968      1969      1970      1971      1972
## 70.24969 70.21302 70.53020 70.21159 70.48781 70.74541 71.10667 71.18100
##      1973      1974      1975      1976      1977      1978      1979      1980
## 71.40062 71.96307 72.54790 72.84992 73.23365 73.41519 73.83803 73.73646
##      1981      1982      1983      1984      1985      1986      1987      1988
## 74.12391 74.47290 74.56344 74.68577 74.66628 74.74541 74.87681 74.85322
##      1989      1990      1991      1992      1993      1994      1995      1996
## 75.12774 75.39754 75.54871 75.80521 75.59326 75.76505 75.87976 76.20107
```



```
##      1997      1998      1999      2000      2001      2002      2003      2004
## 76.52696 76.68826 76.69989 76.83992 76.95717 77.03058 77.18219 77.59121
##      2005      2006      2007      2008      2009      2010      2011      2012
## 77.59102 77.87434 78.13842 78.22962 78.59583 78.80024 78.84706 78.96966
##      2013      2014      2015      2016      2017
## 78.97789 79.05115 78.89756 78.87574 78.85020
```

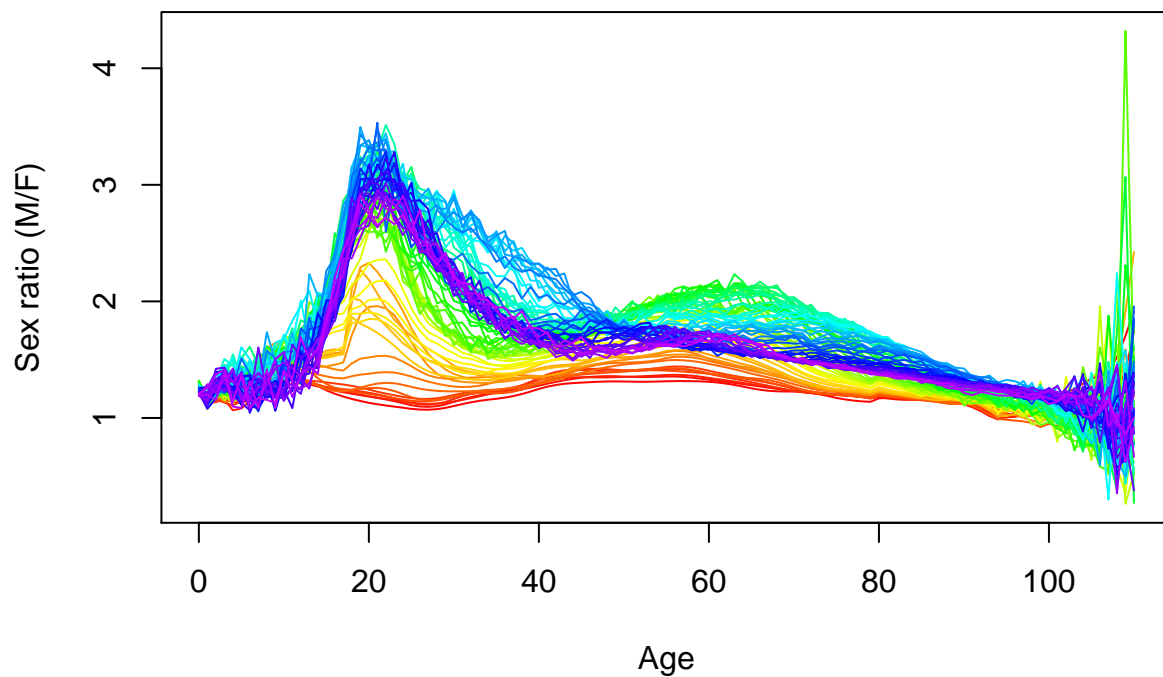
Devuelve la esperanza de vida al nacimiento para cada año que hay en la base. Si se quiere algo más preciso, como la esperanza de vida a los 25 años de mujeres en 1945:

```
espe_vida_mujeres_25_1945<-life.expectancy(USA_mortalidad, series = "Female", age = 25, years = 1945)
espe_vida_mujeres_25_1945
```

```
## Time Series:
## Start = 1945
## End = 1945
## Frequency = 1
## [1] 47.56695
```

La función `sexratio` muestra las diferencias en la mortalidad a diferentes edades:

```
IM<-sex.ratio(USA_mortalidad)
plot(IM)
```



Que muestra el índice de masculinidad de las tasas de mortalidad para diferentes edades, donde cada línea representa una cohorte.

Example Two: Lee Carter Model Lee Carter Models allow us to extend our models to the future and predict mortality or life expectancy. While I won't go too in depth here, it is of note in this package as not many statistical programs have a built in means to perform this task. In order to forecast ahead, our data must first be converted into a Lee-Carter model.

Lets examine the projected total death rates and then project this into the future The `lca` command requires a `demogdata` object as its primary input. With many of the other components mirroring what we have seen above. The `forecast.lca` command utilizes the `lca` output, and allows you to forecast ahead by the number of years indicated in `h`. The `se` and `jumpchoice` arguments refer to methods of computation which can be examined in more depth on the demography package page. The confidence level for prediction intervals is given by the `level` argument.

```
lca2=lca(dat,series=names(datrate)[3],years = datyear,ages=dat$age,max.age=100) forecast.lca(lca2,
h=50, se=c("innovdrift"), jumpchoice=c("fit"), level=90) If we plot our forecasted data we see a similar
plotting mechanism as to our initial data plot.
```

Figure Five: Projected combined male and female death rates for the years 2017 to 2066

```
summary(lca2) ## Lee-Carter analysis ## ## Call: lca(data = dat, series = names(datrate)[3], years =
datyear,
##
## Call: ages = dat$age, max.age = 100) ## ## Adjustment method: dt ## Region: USA ## Years
in fit: 1933 - 2016 ## Ages in fit: 0 - 100 ## ## Percentage variation explained: 96.2% ## ##
ERROR MEASURES BASED ON MORTALITY RATES ## ## Averages across ages: ## ME MSE
MPE MAPE ## -0.00001 0.00005 0.00755 0.06977 ## ## Averages across years: ## IE ISE IPE IAPE
## 0.00005 0.00386 0.75703 6.93471 ## ## ## ERROR MEASURES BASED ON LOG MORTALITY
RATES ## ## Averages across ages: ## ME MSE MPE MAPE ## 0.00281 0.00949 -0.00013 0.01660 ##
## Averages across years: ## IE ISE IPE IAPE ## 0.28073 0.94279 -0.01769 1.60972 Other commands
mean.demogdata Allows for the computation of the mean or median for the demographic rates at each age
level in the supplied demogdata files. Arguments allow for a transformation to be applied ot the data prior
to calculation. extract.years Allows for a vector of values for a specific year or series of years to be extracted
from a demogdata object, creating a new specific demogdata object extract.ages Similar to the above but
for ages, if the last argument is true, all the ages beyond the queried age are combined into the highest age
group given.
```

Biblio

Human Mortality Database. University of California, Berkeley (USA), and Max Planck Institute for Demographic Research (Germany). Available at www.mortality.org or www.humanmortality.de (data downloaded on [date]).