Statistics for life insurance with R

Ana María Pérez-Marín[1,2]

Miguel Santolino[1,2]

[1] Departament d'Econometria, Estadística i Economia Aplicada, Universitat de Barcelona (UB), Av. Diagonal, 690, 08034 Barcelona, Spain.
[2] RISKcenter, Institut de Recerca en Economia Aplicada (IREA), Universitat de Barcelona (UB), Av. Diagonal, 690, 08034 Barcelona, Spain.

Statistics for life insurance with R

Ana María Pérez-Marín and Miguel Santolino

Table of Contents

1. Introduction to the `lifecontingencies` R package

The `lifecontingencies` R package (Spedicato, 2013) is an actuarial package that performs life insurance statistical and mathematical calculations. Here, we present a summary of the most important functions of this package with many examples and exercises (with solutions).

1.1. Creating `lifetable` objects

In order to calculate survival and death probabilities with `lifecontingencies`, the first thing that we have to do is to install and load the `lifecontingencies` R package:

```
#install.packages("lifecontingencies")
library(lifecontingencies)

## Package:  lifecontingencies
## Authors:  Giorgio Alfredo Spedicato [aut, cre]
##     (<https://orcid.org/0000-0002-0315-8888>),
##   Christophe Dutang [ctb] (<https://orcid.org/0000-0001-6732-1501>),
##   Reinhold Kainhofer [ctb] (<https://orcid.org/0000-0002-7895-1311>),
##   Kevin J Owens [ctb],
##   Ernesto Schirmacher [ctb],
##   Gian Paolo Clemente [ctb] (<https://orcid.org/0000-0001-6795-4595>),
##   Ivan Williams [ctb]
## Version:  1.3.8
## Date:     2022-01-07 13:53:00 UTC
## BugReport: https://github.com/spedygiorgio/lifecontingencies/issues
```

Then, the second step is to create a `lifetable` object, that will contain the life table that we are going to use. Life tables collect the annual probabilities of survival and/or death by age and gender, as well as additional biometric functions, such as the life expectancy. To create a `lifetable` we need two basic elements:

1) A sequency of ages 0, 1 ,... ω, where ω is the terminal age.
2) The vector l_x (number of individuals alive at the beginning of age x), or survival/mortality rates (also by age).

There are three ways how to create a `lifetable`:

1) Directly from x and vector l_x.
2) Importing x and l_x from an existing `data.frame`.
3) By using raw mortality/survival probabilities.

Lets see some examples of each of them:

1) Directly from x and vector l_x

```
x1<-seq(from=0, to=19, by =1)
lx1<-c(1000,965,950,914,850,768,740,705,680,634,600,586,550,487,400,367,3
10,256,139,50)
Lt<-new("lifetable",x=x1,lx=lx1,name="Example of life table")
print(Lt)

## Life table Example of life table
##
##     x   lx        px         ex
## 1   0 1000 0.9650000 10.9510000
## 2   1  965 0.9844560 10.3481865
```

```
## 3   2  950 0.9621053  9.5115789
## 4   3  914 0.9299781  8.8862144
## 5   4  850 0.9035294  8.5552941
## 6   5  768 0.9635417  8.4687500
## 7   6  740 0.9527027  7.7891892
## 8   7  705 0.9645390  7.1758865
## 9   8  680 0.9323529  6.4397059
## 10  9  634 0.9463722  5.9069401
## 11 10  600 0.9766667  5.2416667
## 12 11  586 0.9385666  4.3668942
## 13 12  550 0.8854545  3.6527273
## 14 13  487 0.8213552  3.1252567
## 15 14  400 0.9175000  2.8050000
## 16 15  367 0.8446866  2.0572207
## 17 16  310 0.8258065  1.4354839
## 18 17  256 0.5429688  0.7382812
## 19 18  139 0.3597122  0.3597122
```

We see that p_x is obtained by calculating l_{x+1}/l_x, therefore p_0 = 965/1000 = 0.965, and so on. The last column is the residual life expectancy (column e_x, that represent the expected number of years that a person aged x is going to survive). It can be obtained as:

e_0 = (965+950+…+50)/1000 = 10.951,

e_1 = (950+914+…+50)/965 = 10.348,

and so on..

2) By using an existing `data.frame` (the most frequent situation). The `lifecontingencies` package has some `data.frames` that let us to create life tables (see more details in Spedicato, 2013):

Data	Description
AF92Lt	Life table UK AF92 (United Kingdom, Females, 1992)
AM92Lt	Life table UK AM92 (United Kingdom, Males, 1992)
demoChina	Mortality rates of China downloaded from the website of SOA (Society of Actuaries)
demoIta	Italian life tables, such as RG48 or IPS55
demoJapan	Mortality rates of Japan downloaded from the website of SOA
demoUsa	US Social Security life tables
demoFrance	French life tables, 1990 and 2002
soa08	Illustrative SOA life table
soa08Act	Illustrative SOA actuarial table

We create a `lifetable` from `demoUsa`:

```r
data("demoUsa", package="lifecontingencies")
is.data.frame(demoUsa)

## [1] TRUE

US07Male<-demoUsa[,c("age","USSS2007M")]
names(US07Male)<-c("x","lx")
US07MaleLt<-as(US07Male,"lifetable")
print(US07MaleLt)

## Life table COERCED
##
##       x     lx        px        ex
## 1     0 100000 0.9926200 74.881620
## 2     1  99262 0.9995064 74.438355
## 3     2  99213 0.9996875 73.475119
## 4     3  99182 0.9997580 72.498084
## 5     4  99158 0.9997983 71.515632
## 6     5  99138 0.9998184 70.530059
## 7     6  99120 0.9998386 69.542867
## 8     7  99104 0.9998486 68.554095
## 9     8  99089 0.9998587 67.564472
## 10    9  99075 0.9998991 66.574020
## 11   10  99065 0.9999092 65.580740
## 12   11  99056 0.9999091 64.586698
## 13   12  99047 0.9998486 63.592567
## 14   13  99032 0.9997476 62.602199
## 15   14  99007 0.9996061 61.618007
## 16   15  98968 0.9994342 60.642288
## 17   16  98912 0.9992822 59.676622
## 18   17  98841 0.9991198 58.719489
## 19   18  98754 0.9989874 57.771219
## 20   19  98654 0.9988546 56.829779
## 21   20  98541 0.9987112 55.894947
## 22   21  98414 0.9985876 54.967078
## 23   22  98275 0.9985042 54.044823
## 24   23  98128 0.9984918 53.125785
## 25   24  97980 0.9985099 52.206032
## 26   25  97834 0.9985588 51.283940
## 27   26  97693 0.9985874 50.357958
## 28   27  97555 0.9986059 49.429194
## 29   28  97419 0.9986142 48.498199
## 30   29  97284 0.9985918 47.565499
## 31   30  97147 0.9985795 46.632577
## 32   31  97009 0.9985465 45.698915
## 33   32  96868 0.9985134 44.765433
## 34   33  96724 0.9984699 43.832079
## 35   34  96576 0.9984158 42.899250
## 36   35  96423 0.9983510 41.967321
## 37   36  96264 0.9982548 41.036639
## 38   37  96096 0.9981581 40.108381
```

```
## 39   38  95919 0.9980192 39.182393
## 40   39  95729 0.9978690 38.260161
## 41   40  95525 0.9976760 37.341869
## 42   41  95303 0.9974712 36.428853
## 43   42  95062 0.9972439 35.521207
## 44   43  94800 0.9970148 34.619378
## 45   44  94517 0.9967413 33.723034
## 46   45  94209 0.9964547 32.833286
## 47   46  93875 0.9961438 31.950104
## 48   47  93513 0.9957974 31.073787
## 49   48  93120 0.9953930 30.204929
## 50   49  92691 0.9949618 29.344726
## 51   50  92224 0.9944917 28.493321
## 52   51  91716 0.9939923 27.651140
## 53   52  91165 0.9934953 26.818264
## 54   53  90572 0.9930221 25.993850
## 55   54  89940 0.9925506 25.176507
## 56   55  89270 0.9920242 24.365464
## 57   56  88558 0.9914406 23.561361
## 58   57  87800 0.9908314 22.764772
## 59   58  86995 0.9901489 21.975424
## 60   59  86138 0.9894239 21.194061
## 61   60  85227 0.9885834 20.420606
## 62   61  84254 0.9876920 19.656432
## 63   62  83217 0.9867094 18.901378
## 64   63  82111 0.9856779 18.155972
## 65   64  80935 0.9845432 17.419781
## 66   65  79684 0.9832714 16.693263
## 67   66  78351 0.9818509 15.977269
## 68   67  76929 0.9802675 15.272602
## 69   68  75411 0.9785310 14.580035
## 70   69  73792 0.9766099 13.899921
## 71   70  72066 0.9744262 13.232828
## 72   71  70223 0.9719608 12.580123
## 73   72  68254 0.9693351 11.943036
## 74   73  66161 0.9665362 11.320854
## 75   74  63947 0.9634854 10.712809
## 76   75  61612 0.9599916 10.118808
## 77   76  59147 0.9560079  9.540518
## 78   77  56545 0.9516491  8.979538
## 79   78  53811 0.9468510  8.435766
## 80   79  50951 0.9415713  7.909285
## 81   80  47974 0.9355484  7.400092
## 82   81  44882 0.9287242  6.909897
## 83   82  41683 0.9212629  6.440203
## 84   83  38401 0.9130752  5.990625
## 85   84  35063 0.9040584  5.560933
## 86   85  31699 0.8940661  5.151077
## 87   86  28341 0.8829611  4.761406
## 88   87  25024 0.8705643  4.392543
```

```
## 89   88  21785 0.8570117  4.045628
## 90   89  18670 0.8420996  3.720621
## 91   90  15722 0.8259763  3.418267
## 92   91  12986 0.8086401  3.138457
## 93   92  10501 0.7901152  2.881154
## 94   93   8297 0.7705195  2.646499
## 95   94   6393 0.7498827  2.434694
## 96   95   4794 0.7292449  2.246767
## 97   96   3496 0.7090961  2.080950
## 98   97   2479 0.6901977  1.934651
## 99   98   1711 0.6715371  1.803039
## 100  99   1149 0.6562228  1.684943
## 101 100    754 0.6379310  1.567639
## 102 101    481 0.6195426  1.457380
## 103 102    298 0.6006711  1.352349
## 104 103    179 0.5810056  1.251397
## 105 104    104 0.5576923  1.153846
## 106 105     58 0.5344828  1.068966
## 107 106     31 0.5161290  1.000000
## 108 107     16 0.5000000  0.937500
## 109 108      8 0.5000000  0.875000
## 110 109      4 0.5000000  0.750000
## 111 110      2 0.5000000  0.500000
```

```
US07MaleLt@name<-"USA MALES 2007"
getOmega(US07MaleLt)

## [1] 111
```

The function getOmega returns the terminal age of the life table, that is to say, the maximum age of the life table.

3) By using raw survivial/mortality probabilities

```
#we create a vector with the survival probabilities

probsup<-seq(0.9,0,by=-0.01)
probsup

##  [1] 0.90 0.89 0.88 0.87 0.86 0.85 0.84 0.83 0.82 0.81 0.80 0.79 0.78
0.77 0.76
## [16] 0.75 0.74 0.73 0.72 0.71 0.70 0.69 0.68 0.67 0.66 0.65 0.64 0.63
0.62 0.61
## [31] 0.60 0.59 0.58 0.57 0.56 0.55 0.54 0.53 0.52 0.51 0.50 0.49 0.48
0.47 0.46
## [46] 0.45 0.44 0.43 0.42 0.41 0.40 0.39 0.38 0.37 0.36 0.35 0.34 0.33
0.32 0.31
## [61] 0.30 0.29 0.28 0.27 0.26 0.25 0.24 0.23 0.22 0.21 0.20 0.19 0.18
0.17 0.16
## [76] 0.15 0.14 0.13 0.12 0.11 0.10 0.09 0.08 0.07 0.06 0.05 0.04 0.03
0.02 0.01
## [91] 0.00
```

```r
#with the function probs2lifetable we create the life table

LifeT<-probs2lifetable(probsup,type="px",name="Life table obtained by usi
ng raw probabilities")
LifeT

## Life table Life table obtained by using raw probabilities
##
##     x           lx   px          ex
## 1   0 1.000000e+04 0.90 5.84474214
## 2   1 9.000000e+03 0.89 5.49415793
## 3   2 8.010000e+03 0.88 5.17321116
## 4   3 7.048800e+03 0.87 4.87864904
## 5   4 6.132456e+03 0.86 4.60764258
## 6   5 5.273912e+03 0.85 4.35772393
## 7   6 4.482825e+03 0.84 4.12673403
## 8   7 3.765573e+03 0.83 3.91277861
## 9   8 3.125426e+03 0.82 3.71419110
## 10  9 2.562849e+03 0.81 3.52950134
## 11 10 2.075908e+03 0.80 3.35740906
## 12 11 1.660726e+03 0.79 3.19676132
## 13 12 1.311974e+03 0.78 3.04653332
## 14 13 1.023340e+03 0.77 2.90581194
## 15 14 7.879714e+02 0.76 2.77378175
## 16 15 5.988583e+02 0.75 2.64971282
## 17 16 4.491437e+02 0.74 2.53295043
## 18 17 3.323664e+02 0.73 2.42290599
## 19 18 2.426274e+02 0.72 2.31904930
## 20 19 1.746918e+02 0.71 2.22090181
## 21 20 1.240311e+02 0.70 2.12803072
## 22 21 8.682180e+01 0.69 2.04004388
## 23 22 5.990704e+01 0.68 1.95658533
## 24 23 4.073679e+01 0.67 1.87733137
## 25 24 2.729365e+01 0.66 1.80198712
## 26 25 1.801381e+01 0.65 1.73028351
## 27 26 1.170898e+01 0.64 1.66197464
## 28 27 7.493744e+00 0.63 1.59683537
## 29 28 4.721059e+00 0.62 1.53465932
## 30 29 2.927057e+00 0.61 1.47525697
## 31 30 1.785504e+00 0.60 1.41845405
## 32 31 1.071303e+00 0.59 1.36409008
## 33 32 6.320686e-01 0.58 1.31201709
## 34 33 3.665998e-01 0.57 1.26209843
## 35 34 2.089619e-01 0.56 1.21420776
## 36 35 1.170186e-01 0.55 1.16822815
## 37 36 6.436026e-02 0.54 1.12405118
## 38 37 3.475454e-02 0.53 1.08157626
## 39 38 1.841991e-02 0.52 1.04070992
## 40 39 9.578351e-03 0.51 1.00136524
## 41 40 4.884959e-03 0.50 0.96346125
```

```
## 42 41 2.442479e-03 0.49 0.92692249
## 43 42 1.196815e-03 0.48 0.89167855
## 44 43 5.744712e-04 0.47 0.85766365
## 45 44 2.700015e-04 0.46 0.82481628
## 46 45 1.242007e-04 0.45 0.79307887
## 47 46 5.589030e-05 0.44 0.76239748
## 48 47 2.459173e-05 0.43 0.73272155
## 49 48 1.057444e-05 0.42 0.70400360
## 50 49 4.441267e-06 0.41 0.67619904
## 51 50 1.820919e-06 0.40 0.64926596
## 52 51 7.283678e-07 0.39 0.62316489
## 53 52 2.840634e-07 0.38 0.59785869
## 54 53 1.079441e-07 0.37 0.57331234
## 55 54 3.993932e-08 0.36 0.54949282
## 56 55 1.437815e-08 0.35 0.52636894
## 57 56 5.032354e-09 0.34 0.50391125
## 58 57 1.711000e-09 0.33 0.48209191
## 59 58 5.646301e-10 0.32 0.46088459
## 60 59 1.806816e-10 0.31 0.44026434
## 61 60 5.601131e-11 0.30 0.42020754
## 62 61 1.680339e-11 0.29 0.40069181
## 63 62 4.872984e-12 0.28 0.38169591
## 64 63 1.364435e-12 0.27 0.36319966
## 65 64 3.683976e-13 0.26 0.34518394
## 66 65 9.578337e-14 0.25 0.32763054
## 67 66 2.394584e-14 0.24 0.31052216
## 68 67 5.747002e-15 0.23 0.29384235
## 69 68 1.321811e-15 0.22 0.27757542
## 70 69 2.907983e-16 0.21 0.26170646
## 71 70 6.106765e-17 0.20 0.24622124
## 72 71 1.221353e-17 0.19 0.23110619
## 73 72 2.320571e-18 0.18 0.21634838
## 74 73 4.177027e-19 0.17 0.20193545
## 75 74 7.100946e-20 0.16 0.18785561
## 76 75 1.136151e-20 0.15 0.17409759
## 77 76 1.704227e-21 0.14 0.16065061
## 78 77 2.385918e-22 0.13 0.14750437
## 79 78 3.101693e-23 0.12 0.13464901
## 80 79 3.722032e-24 0.11 0.12207509
## 81 80 4.094235e-25 0.10 0.10977358
## 82 81 4.094235e-26 0.09 0.09773581
## 83 82 3.684811e-27 0.08 0.08595349
## 84 83 2.947849e-28 0.07 0.07441866
## 85 84 2.063494e-29 0.06 0.06312367
## 86 85 1.238097e-30 0.05 0.05206121
## 87 86 6.190483e-32 0.04 0.04122424
## 88 87 2.476193e-33 0.03 0.03060600
## 89 88 7.428580e-35 0.02 0.02020000
## 90 89 1.485716e-36 0.01 0.01000000
```

1.2. Creating an `actuarialtable` object

Once we create a `lifetable`, we can also create an `actuarialtable`. An `actuarialtable` is similar to a `lifetable`, but it also incorporates an interest rate that will be necessary in order to calculate the actuarial present value (APV) of life insurance products. As an example, here we show how to create an `actuarialtable` by using an interest rate of 3 percent. Columns D_x, N_x, C_x and M_x are specific functions of l_x and the interest rate (see more details in Spedicato, 2013):

```
US07MaleAct<-new("actuarialtable",x=US07MaleLt@x,lx=US07MaleLt@lx,interes
t=0.03,name="USA MALES 2007 ACT TABLE")
print(US07MaleAct)

## Actuarial table  USA MALES 2007 ACT TABLE interest rate  3 %
##
##        x     lx           Dx           Nx           Cx           Mx
Rx
## 1      0 100000 1.000000e+05 3.000556e+06 716.50485437 1.260516e+04 7.8
64341e+05
## 2      1  99262 9.637087e+04 2.900556e+06  46.18719955 1.188865e+04 7.7
38290e+05
## 3      2  99213 9.351777e+04 2.804185e+06  28.36939144 1.184247e+04 7.6
19403e+05
## 4      3  99182 9.076558e+04 2.710668e+06  21.32368915 1.181410e+04 7.5
00979e+05
## 5      4  99158 8.810060e+04 2.619902e+06  17.25217569 1.179277e+04 7.3
82838e+05
## 6      5  99138 8.551731e+04 2.531801e+06  15.07471662 1.177552e+04 7.2
64910e+05
## 7      6  99120 8.301144e+04 2.446284e+06  13.00946418 1.176045e+04 7.1
47155e+05
## 8      7  99104 8.058062e+04 2.363273e+06  11.84113851 1.174744e+04 7.0
29550e+05
## 9      8  99089 7.822177e+04 2.282692e+06  10.72983425 1.173560e+04 6.9
12076e+05
## 10     9  99075 7.593274e+04 2.204470e+06   7.44093915 1.172487e+04 6.7
94720e+05
## 11    10  99065 7.371366e+04 2.128538e+06   6.50179149 1.171742e+04 6.6
77471e+05
## 12    11  99056 7.156016e+04 2.054824e+06   6.31241892 1.171092e+04 6.5
60297e+05
## 13    12  99047 6.946957e+04 1.983264e+06  10.21427010 1.170461e+04 6.4
43188e+05
## 14    13  99032 6.743597e+04 1.913794e+06  16.52794515 1.169440e+04 6.3
26142e+05
## 15    14  99007 6.545529e+04 1.846358e+06  25.03261595 1.167787e+04 6.2
09198e+05
## 16    15  98968 6.352379e+04 1.780903e+06  34.89734860 1.165284e+04 6.0
92419e+05
## 17    16  98912 6.163869e+04 1.717379e+06  42.95616765 1.161794e+04 5.9
```

```
75891e+05
## 18   17   98841 5.980043e+04 1.655740e+06  51.10333086 1.157498e+04 5.8
59711e+05
## 19   18   98754 5.800757e+04 1.595940e+06  57.02860268 1.152388e+04 5.7
43961e+05
## 20   19   98654 5.626100e+04 1.537932e+06  62.56536022 1.146685e+04 5.6
28723e+05
## 21   20   98541 5.455976e+04 1.481671e+06  68.26875804 1.140428e+04 5.5
14054e+05
## 22   21   98414 5.290237e+04 1.427112e+06  72.54305762 1.133602e+04 5.4
00011e+05
## 23   22   98275 5.128899e+04 1.374209e+06  74.48368702 1.126347e+04 5.2
86651e+05
## 24   23   98128 4.972065e+04 1.322920e+06  72.80619298 1.118899e+04 5.1
74016e+05
## 25   24   97980 4.819967e+04 1.273200e+06  69.73041311 1.111618e+04 5.0
62127e+05
## 26   25   97834 4.672606e+04 1.225000e+06  65.38095657 1.104645e+04 4.9
50965e+05
## 27   26   97693 4.529973e+04 1.178274e+06  62.12608970 1.098107e+04 4.8
40500e+05
## 28   27   97555 4.391819e+04 1.132974e+06  59.44243843 1.091895e+04 4.7
30689e+05
## 29   28   97419 4.257958e+04 1.089056e+06  57.28675891 1.085950e+04 4.6
21500e+05
## 30   29   97284 4.128211e+04 1.046476e+06  56.44218605 1.080222e+04 4.5
12905e+05
## 31   30   97147 4.002328e+04 1.005194e+06  55.19822603 1.074577e+04 4.4
04883e+05
## 32   31   97009 3.880235e+04 9.651710e+05  54.75552181 1.069058e+04 4.2
97425e+05
## 33   32   96868 3.761743e+04 9.263687e+05  54.29177953 1.063582e+04 4.1
90519e+05
## 34   33   96724 3.646749e+04 8.887512e+05  54.17464516 1.058153e+04 4.0
84161e+05
## 35   34   96576 3.535115e+04 8.522837e+05  54.37365986 1.052735e+04 3.9
78346e+05
## 36   35   96423 3.426713e+04 8.169326e+05  54.86015558 1.047298e+04 3.8
73072e+05
## 37   36   96264 3.321420e+04 7.826655e+05  56.27713341 1.041812e+04 3.7
68343e+05
## 38   37   96096 3.219052e+04 7.494513e+05  57.56502897 1.036184e+04 3.6
64161e+05
## 39   38   95919 3.119537e+04 7.172607e+05  59.99317374 1.030428e+04 3.5
60543e+05
## 40   39   95729 3.022677e+04 6.860654e+05  62.53759552 1.024428e+04 3.4
57500e+05
## 41   40   95525 2.928384e+04 6.558386e+05  66.07341617 1.018175e+04 3.3
55057e+05
## 42   41   95303 2.836484e+04 6.265548e+05  69.63917299 1.011567e+04 3.2
```

```
53240e+05
## 43   42  95062 2.746904e+04 5.981899e+05  73.50224922 1.004603e+04 3.1
52083e+05
## 44   43  94800 2.659547e+04 5.707209e+05  77.08121444 9.972532e+03 3.0
51623e+05
## 45   44  94517 2.574376e+04 5.441254e+05  81.44709612 9.895451e+03 2.9
51897e+05
## 46   45  94209 2.491250e+04 5.183816e+05  85.75000033 9.814004e+03 2.8
52943e+05
## 47   46  93875 2.410114e+04 4.934691e+05  90.23167292 9.728254e+03 2.7
54803e+05
## 48   47  93513 2.330893e+04 4.693680e+05  95.10552877 9.638022e+03 2.6
57520e+05
## 49   48  93120 2.253493e+04 4.460591e+05 100.79367535 9.542917e+03 2.5
61140e+05
## 50   49  92691 2.177778e+04 4.235241e+05 106.52600626 9.442123e+03 2.4
65711e+05
## 51   50  92224 2.103695e+04 4.017464e+05 112.50329761 9.335597e+03 2.3
71290e+05
## 52   51  91716 2.031172e+04 3.807094e+05 118.47205294 9.223094e+03 2.2
77934e+05
## 53   52  91165 1.960164e+04 3.603977e+05 123.78892286 9.104622e+03 2.1
85703e+05
## 54   53  90572 1.890693e+04 3.407960e+05 128.08755751 8.980833e+03 2.0
94657e+05
## 55   54  89940 1.822816e+04 3.218891e+05 131.83400444 8.852745e+03 2.0
04848e+05
## 56   55  89270 1.756541e+04 3.036610e+05 136.01769477 8.720911e+03 1.9
16321e+05
## 57   56  88558 1.691777e+04 2.860956e+05 140.58772313 8.584894e+03 1.8
29112e+05
## 58   57  87800 1.628444e+04 2.691778e+05 144.95621733 8.444306e+03 1.7
43263e+05
## 59   58  86995 1.566518e+04 2.528933e+05 149.82509589 8.299350e+03 1.6
58820e+05
## 60   59  86138 1.505908e+04 2.372282e+05 154.62684500 8.149525e+03 1.5
75826e+05
## 61   60  85227 1.446584e+04 2.221691e+05 160.34009377 7.994898e+03 1.4
94331e+05
## 62   61  84254 1.388417e+04 2.077032e+05 165.90933580 7.834558e+03 1.4
14382e+05
## 63   62  83217 1.331386e+04 1.938191e+05 171.79478274 7.668648e+03 1.3
36036e+05
## 64   63  82111 1.275429e+04 1.805052e+05 177.34744685 7.496854e+03 1.2
59350e+05
## 65   64  80935 1.220546e+04 1.677509e+05 183.16298132 7.319506e+03 1.1
84381e+05
## 66   65  79684 1.166679e+04 1.555455e+05 189.48433804 7.136343e+03 1.1
11186e+05
## 67   66  78351 1.113750e+04 1.438787e+05 196.24813632 6.946859e+03 1.0
```

```
39823e+05
## 68   67  76929 1.061686e+04 1.327412e+05 203.39510258 6.750611e+03 9.7
03542e+04
## 69   68  75411 1.010423e+04 1.221243e+05 210.60968768 6.547216e+03 9.0
28481e+04
## 70   69  73792 9.599327e+03 1.120201e+05 217.98924239 6.336606e+03 8.3
73760e+04
## 71   70  72066 9.101746e+03 1.024208e+05 225.98644024 6.118617e+03 7.7
40099e+04
## 72   71  70223 8.610660e+03 9.331901e+04 234.40428008 5.892630e+03 7.1
28238e+04
## 73   72  68254 8.125459e+03 8.470835e+04 241.90888786 5.658226e+03 6.5
38975e+04
## 74   73  66161 7.646887e+03 7.658289e+04 248.44083965 5.416317e+03 5.9
73152e+04
## 75   74  63947 7.175721e+03 6.893601e+04 254.38706931 5.167876e+03 5.4
31520e+04
## 76   75  61612 6.712332e+03 6.176029e+04 260.72810372 4.913489e+03 4.9
14733e+04
## 77   76  59147 6.256099e+03 5.504795e+04 267.20279087 4.652761e+03 4.4
23384e+04
## 78   77  56545 5.806680e+03 4.879185e+04 272.58062515 4.385558e+03 3.9
58108e+04
## 79   78  53811 5.364973e+03 4.298517e+04 276.83773124 4.112978e+03 3.5
19552e+04
## 80   79  50951 4.931874e+03 3.762020e+04 279.76981665 3.836140e+03 3.1
08254e+04
## 81   80  47974 4.508457e+03 3.268833e+04 282.11376967 3.556370e+03 2.7
24640e+04
## 82   81  44882 4.095029e+03 2.817987e+04 283.37518343 3.274256e+03 2.3
69003e+04
## 83   82  41683 3.692381e+03 2.408484e+04 282.25973287 2.990881e+03 2.0
41577e+04
## 84   83  38401 3.302576e+03 2.039246e+04 278.71443186 2.708621e+03 1.7
42489e+04
## 85   84  35063 2.927670e+03 1.708988e+04 272.70423798 2.429907e+03 1.4
71627e+04
## 86   85  31699 2.569694e+03 1.416221e+04 264.28917007 2.157203e+03 1.2
28636e+04
## 87   86  28341 2.230560e+03 1.159252e+04 253.45853609 1.892914e+03 1.0
12916e+04
## 88   87  25024 1.912133e+03 9.361959e+03 240.28971038 1.639455e+03 8.2
36249e+03
## 89   88  21785 1.616150e+03 7.449825e+03 224.35980415 1.399165e+03 6.5
96794e+03
## 90   89  18670 1.344718e+03 5.833675e+03 206.14711236 1.174805e+03 5.1
97629e+03
## 91   90  15722 1.099405e+03 4.488956e+03 185.74992406 9.686583e+02 4.0
22823e+03
## 92   91  12986 8.816332e+02 3.389552e+03 163.79540726 7.829084e+02 3.0
```

```
54165e+03
## 93   92  10501 6.921592e+02 2.507918e+03 141.04240105 6.191130e+02 2.2
71256e+03
## 94   93   8297 5.309568e+02 1.815759e+03 118.29539037 4.780706e+02 1.6
52143e+03
## 95   94   6393 3.971967e+02 1.284802e+03  96.45219528 3.597752e+02 1.1
74073e+03
## 96   95   4794 2.891756e+02 8.876058e+02  76.01531872 2.633230e+02 8.1
42975e+02
## 97   96   3496 2.047377e+02 5.984302e+02  57.82426970 1.873077e+02 5.5
09744e+02
## 98   97   2479 1.409502e+02 3.936925e+02  42.39485936 1.294834e+02 3.6
36667e+02
## 99   98   1711 9.445001e+01 2.527423e+02  30.11972968 8.708858e+01 2.3
41833e+02
## 100  99   1149 6.157930e+01 1.582923e+02  20.55297174 5.696885e+01 1.4
70947e+02
## 101 100    754 3.923276e+01 9.671298e+01  13.79122843 3.641588e+01 9.0
12586e+01
## 102 101    481 2.429883e+01 5.748022e+01   8.97540738 2.262465e+01 5.3
70998e+01
## 103 102    298 1.461569e+01 3.318139e+01   5.66647291 1.364924e+01 3.1
08533e+01
## 104 103    179 8.523518e+00 1.856570e+01   3.46728782 7.982770e+00 1.7
43609e+01
## 105 104    104 4.807972e+00 1.004218e+01   2.06466330 4.515482e+00 9.4
53317e+00
## 106 105     58 2.603271e+00 5.234209e+00   1.17657047 2.450818e+00 4.9
37835e+00
## 107 106     31 1.350877e+00 2.630938e+00   0.63461191 1.274248e+00 2.4
87017e+00
## 108 107     16 6.769194e-01 1.280061e+00   0.32860163 6.396360e-01 1.2
12769e+00
## 109 108      8 3.286016e-01 6.031413e-01   0.15951536 3.110344e-01 5.7
31327e-01
## 110 109      4 1.595154e-01 2.745396e-01   0.07743464 1.515191e-01 2.6
20983e-01
## 111 110      2 7.743464e-02 1.150243e-01   0.03758963 7.408442e-02 1.1
05792e-01
## 112 111      1 3.758963e-02 3.758963e-02   0.03649479 3.649479e-02 3.6
49479e-02
```

Note that it is possible to create a `data.frame` from an `actuarialtable`, simply by doing the following:

```
US07MaleActDf<-as(US07MaleAct,"data.frame")
```

1.3. Basic functions for the calculation of probabilities

In this table there are the basic functions available in `lifecontingencies` for calculating survival and mortality probabilities (you can check the complete list of functions in Spedicato, 2013).

Function	Description
dxt	number of deaths between x and x+t
pxt	survival probability of an individual aged x between x and x+t
qxt	probability of death for an individual aged x between x and x+t
mxt	central mortality rate
exn	life expectancy between x and x+n

Here there are some examples, where we use the `lifetable` `US07MaleLt`:

```
# probability that an individual aged 20 survives until 21
pxt(US07MaleLt,20,1)

## [1] 0.9987112

# probability that an individual aged 30 dies in next 2 years
qxt(US07MaleLt,30,2)

## [1] 0.002871936

# probability that an individual aged 30 dies between age 33 and 35
pxt(US07MaleLt,30,3)*qxt(US07MaleLt,33,2)

## [1] 0.003098397
```

1.4. Assumptions for fractional ages

The functions `pxt`, `pxyzt`, `qxt` and `qxyzt` can be used with a specific argument called `fractional` (that can be equal to `linear`, `hyperbolic` or `constant force`) that let us to calculate probabilities for fractional ages or periods (see more details in Ayuso et al., 2007 and Spedicato, 2013). The assumptions are:

a) Linear: Linear interpolation between consecutive ages (assume a uniform distribution for the yearly deaths). In that case, $l_{x+t} = l_x - t \cdot d_x$, therefore, ${}_tq_x = t \cdot q_x$. This is the default assumption.

b) Hyperbolic: Also called Balducci assumption or harmonic interpolation. In that case, $l_{x+t} = (l_x \cdot l_{x+1})/(t \cdot l_x + (1-t) \cdot l_{x+1})$ and therefore, ${}_tq_x = (t \cdot q_x)/(1-(1-t) \cdot q_x)$.

c) Constant force of mortality: It assumes a constant force of mortality, also known as exponential interpolation. In that case, ${}_tq_x = 1 - exp(-t \cdot \mu_x)$ where

μ_x is the force of mortality, that is assumed to be constant between age x and $x+t$. Based on that, for $t = 0.5$, it can be proved that ${}_tq_x = 1 - \sqrt{p_x}$.

Examples:

```
#example linear
qxt(US07MaleLt,20,t=0.5)

## [1] 0.0006444018

0.5*qxt(US07MaleLt,20,t=1)

## [1] 0.0006444018

#example hyperbolic
qxt(US07MaleLt,20,t=0.5,"hyperbolic")

## [1] 0.0006448173

0.5*qxt(US07MaleLt,20,t=1)/(1-0.5*qxt(US07MaleLt,20,t=1))

## [1] 0.0006448173

#example constant force
qxt(US07MaleLt,20,t=0.5,"constant force")

## [1] 0.0006446096

1-sqrt(pxt(US07MaleLt,20,t=1))

## [1] 0.0006446096
```

1.5. Exercises

Exercise 1: Create a `lifetable` R object called `FranceH02Lt` from the lifetable `demoFrance` (columns age and `TH00_02`) available in the package `lifecontingencies`. The `lifetable` must be called `FranceH02Lt` and must have the label "France Males 2002".

By using this table, you will calculate:

a) Life expectancy of a men aged 35.
b) Probability that an insured aged 35 would die in next 5 years.
c) Probability that an insured aged 35 reaches age 40.
d) Probability that an insured aged 35 dies at the age of 40. Make the calculation in three different ways.
e) Probability that an insured aged 35 dies between 40 and 50 years old.
f) Probability that an insured aged 35 dies in next 9 months (constant force assumption).

```r
#Solution

data(demoFrance)
FranceH02<-demoFrance[,c("age","TH00_02")]
names(FranceH02)<-c("x","lx")
FranceH02Lt<-as(FranceH02,"lifetable")
FranceH02Lt@name<-"France Males 2002"

#a) e35
exn(FranceH02Lt,35)

## [1] 41.58141

#b) 5q35
qxt(FranceH02Lt,x=35,t=5)

## [1] 0.009048936

#c) 5p35
1-qxt(FranceH02Lt,x=35,t=5)

## [1] 0.9909511

# or
pxt(FranceH02Lt,x=35,t=5)

## [1] 0.9909511

#d) 5/q35
pxt(FranceH02Lt,x=35,t=5)*qxt(FranceH02Lt,x=40,t=1)

## [1] 0.002344497

# or
pxt(FranceH02Lt,x=35,t=5)-pxt(FranceH02Lt,x=35,t=6)

## [1] 0.002344497

# or
qxt(FranceH02Lt,x=35,t=6)-qxt(FranceH02Lt,x=35,t=5)

## [1] 0.002344497

#e) 5/10q35
pxt(FranceH02Lt,x=35,t=5)*qxt(FranceH02Lt,x=40,t=10)

## [1] 0.03735771

#f) 0.75q35
qxt(FranceH02Lt,x=35,t=0.75, "constant force")

## [1] 0.001149332
```

Exercise 2: By using the `lifetable` `US07MaleLt` that we have just created, make the following calculations:

a) Probability that an insured aged 55 reaches age 63.

b) Probability that an insured aged 55 dies in next 6 months (linear assumption).

c) Probability that an insured aged 55 dies in his first year of retirement (that starts at age 65).

d) Probability that an insured aged 55 dies in the first three years of retirement.

```
# Solution

#a) 8p55
pxt(US07MaleLt,x=55,t=8)

## [1] 0.9198051

#b) 0.5q55
qxt(US07MaleLt,x=55,t=0.5)

## [1] 0.003987902

#c) 10/q55
pxt(US07MaleLt,x=55,t=10)*qxt(US07MaleLt,x=65,t=1)

## [1] 0.01493223

#d) 10/3q55
pxt(US07MaleLt,x=55,t=10)*qxt(US07MaleLt,x=65,t=3)

## [1] 0.04786602
```

2. Calculating two-lives probabilities with lifecontingencies

This section focuses on the calculation of two-lives survival and death probabilities with `lifecontingencies`. The first thing that we have to do is to create joint life tables.

2.1. Creating joint life tables with lifecontingencies

A joint life table is a list object built with individual life tables (for example, one for man and another for woman). Example:

```
US07Female<-demoUsa[,c("age","USSS2007F")]
names(US07Female)<-c("x","lx")
US07FemaleLt<-as(US07Female,"lifetable") # it is a life table object
US07FemaleLt@name<-"USA FEMALES 2007"
US07List<-list(US07MaleLt,US07FemaleLt) #joint life table
```

2.2. Basic functions for calculating two-lives probabilities

Once the joint life table is created, we can use the following basic functions for calculating two-lives probabilities with `lifecontingencies`:

Función	Descripción
pxyt	joint-life probability (two lives)
pxyzt	joint-life probability (more than two lives)
qxyt	probability of death (two lives)
qxyzt	probability of death (more than two lives)

All these functions have a specific argument called `status`. The `status` let us indicate if we want to calculate the joint-life, last-survivor, etc.. probabilities. Note that the `joint` status exists while all the members of the group are alive, while the `last` status exists until the last member of the group dies. Specifically, we have that:

- `pxyt` with `status = joint` let us calculate the joint-life probability (in the Spanish terminology, called "supervivencia conjunta" or "no disolución").
- `pxyt` with `status = last` let us calculate the non last-survivor probability (in Spanish, called "al menos uno vive" or "no extinción").
- `qxyt` with `status = joint` let us calculate the non joint-life probability (in Spanish called "disolución").
- `qxyt` with `status = last` let us calculate the last-survivor probability (in Spanish called "extinción").

By default, `status` is equal to `joint`.

Examples:

```
# joint-life probability of a couple (man aged 55, woman aged 50) during
next 3 years
pxyzt(US07List,x=c(55,50),t=3)

## [1] 0.9642736

# non last-survivor probability of a couple (man aged 55, woman aged 50)
during next 3 years
pxyzt(US07List,x=c(55,50),t=3, status="last")

## [1] 0.9997322

# non joint-life probability of a couple (man aged 55, woman aged 50) dur
ing next 3 years
qxyzt(US07List,x=c(55,50),t=3)

## [1] 0.03572644
```

```
# last-survivor probability of a couple (man aged 55, woman aged 50) duri
ng next 3 years
qxyzt(US07List,x=c(55,50),t=3, status= "last")

## [1] 0.0002678365
```

It is clear that the non joint-life probability is complementary to the joint-life probability, therefore:

```
1-pxyzt(US07List,x=c(55,50),t=3)

## [1] 0.03572644
```

and

```
qxyzt(US07List,x=c(55,50),t=3)

## [1] 0.03572644
```

give the same result.

On the other hand, the last-survivor probability is complementary to the non last survivor probability, therefore:

```
1-qxyzt(US07List,x=c(55,50),t=3,status = "last")

## [1] 0.9997322
```

equals

```
pxyzt(US07List,x=c(55,50),t=3,status = "last")

## [1] 0.9997322
```

2.3. Exercises

Exercise 1: By using the table `usa07List` calculate the joint-life probability of a couple (man aged 65 and woman aged 63), during next two years. Do the calculation also by using the survival probabilities of man and woman and check that you get the same result.

```
#solution
pxyzt(US07List,x=c(65,63),t=2)

## [1] 0.9472972

pxt(US07MaleLt,65,2)*pxt(US07FemaleLt,63,2) #same result

## [1] 0.9472972
```

Exercise 2: By using the table `usa07List` calculate the non last-survivor probability (at least one is alive) of a couple (man aged 65 and woman aged 63), during next 2 years. Do the calculation in three different ways and check that you get the same result.

```
#solution
pxyzt(US07List,x=c(65,63),t=2,status="last")

## [1] 0.9993508

#we get the same result
pxt(US07MaleLt,65,2)*pxt(US07FemaleLt,63,2)+qxt(US07MaleLt,65,2)*pxt(US07
FemaleLt,63,2)+
  +pxt(US07MaleLt,65,2)*qxt(US07FemaleLt,63,2)

## [1] 0.9993508

#we get the same result
1-qxyzt(US07List,x=c(65,63),t=2,status="last")

## [1] 0.9993508
```

Exercise 3: By using the table `usa07List` calculate the last-survivor probability of a couple (man aged 65 and woman aged 63), during next 2 years. Do the calculation in two different ways and check that you get the same result.

```
#solution
qxyzt(US07List,x=c(65,63),t=2,status="last") #last-survivor

## [1] 0.0006492309

qxt(US07MaleLt,65,t=2)*qxt(US07FemaleLt,63,t=2) #both die, same result

## [1] 0.0006492309
```

Exercise 4: By using the table `usa07List` calculate the non joint-life (at least one die) probability of a couple (man aged 65 and woman aged 63), during next 2 years.

```
#solution
qxyzt(US07List,x=c(65,63),t=2,status="joint")

## [1] 0.05270281

1-pxyzt(US07List,x=c(65,63),t=2) #same result

## [1] 0.05270281
```

Exercise 5: Do the following calculations for a couple (man aged 45, woman aged 42) by using the life table `US07List`:

a) Joint-life probability of the couple during next 12 years.

b) Non joint-life probability (at least one die) of the couple during next 12 years. Do the calculation in two different ways.

c) Last-survivor probability (all die) of the couple during next 12 years. Do the calculation in two different ways.

d) Probability that exactly one member of the couple survives after 12 years. Do the calculation in two different ways.

e) Probability that both members of the couple survive 12 years and after that, at least one of them dies in the following year.

f) Probability that the couple survive 12 years and after that, both of them die in the following year.

g) Deferred last-survivor probability of the couple (deferral = 12 years, period = 1 year). Do the calculation in two different ways.

```
#a) 12p45_42
pxyzt(US07List,x=c(45,42),t=12)

## [1] 0.9016853

#b) 12q45_42
1-pxyzt(US07List,x=c(45,42),t=12)

## [1] 0.09831468

# or
qxyzt(US07List,x=c(45,42),t=12,status="joint")

## [1] 0.09831468

#c) 12q45_42-
qxyzt(US07List,x=c(45,42),t=12,status="last")

## [1] 0.002210673

# or
qxt(US07MaleLt,x=45,t=12)*qxt(US07FemaleLt,x=42,t=12)

## [1] 0.002210673

#d) 12q45_42 [1]
qxyzt(US07List,x=c(45,42),t=12,status="joint")-qxyzt(US07List,x=c(45,42),
t=12,status="last")

## [1] 0.09610401

# or
qxt(US07MaleLt,x=45,t=12)*pxt(US07FemaleLt,x=42,t=12)+
  pxt(US07MaleLt,x=45,t=12)*qxt(US07FemaleLt,x=42,t=12)

## [1] 0.09610401
```

```
#e) 12/q45_42
pxyzt(US07List,x=c(45,42),t=12)*qxyzt(US07List,x=c(45+12,42+12),t=1)

## [1] 0.01212011

#f) 12p45_42 * q57_54-
pqext<-pxyzt(US07List,x=c(45,42),t=12)*qxyzt(US07List,x=c(45+12,42+12),t=
1,status="last")
pqext

## [1] 3.565286e-05

#g) 12/q45_42-
pxyzt(US07List,x=c(45,42),t=12,status="last")-pxyzt(US07List,x=c(45,42),t
=13,status="last")

## [1] 0.0005971736

# or
pqext+qxt(US07MaleLt,x=45,t=12)*pxt(US07FemaleLt,x=42,t=12)*qxt(US07Femal
eLt,x=42+12,t=1)+
  +qxt(US07FemaleLt,x=42,t=12)*pxt(US07MaleLt,x=45,t=12)*qxt(US07MaleLt,x
=45+12,t=1)

## [1] 0.0005971736
```

3. Calculating the actuarial present value APV with lifecontingencies (one life)

In this section, we will see how to calculate the actuarial present value (APV) of life insurance contracts, where the payment of the benefit depends on the survival of just one individual.

3.1. Basic functions for calculating the APV

In the next table, we show the basic functions for calculating the actuarial present value (APV) of life insurance contracts (one life) with `lifecontingencies`.

Funcion	Descripcion
Axn	one-life insurance, pays 1 m.u. at the end of the year of death
AExn	n-year endowment, it pays 1 m.u. at the end of the year of death or after n years, if the insured survives
Exn	pure endowment, it pays 1 m.u. after a period of n years, if the insured survives

Funcion	Descripcion
IAxn	increasing life insurance contract (it starts paying 1 m.u. if death occurs in the first year of coverage and after that, the benefit is increased by 1 m.u. every year)
DAxn	decreasing life insurance (it starts paying n m.u. if death occurs in the first year of coverage and after that, the benefit is decreased by 1 m.u. every year)

Some examples:

Example 1: By using the table `soa08Act` calculate the APV of a contract that pays 1 m.u. at the end of the death year. The insured is 25 years old and the coverage period is 40 years.

```
data(soa08Act,package="lifecontingencies")
Axn(actuarialtable=soa08Act,x=25,n=40)

## [1] 0.04797088
```

Example 2: By using the table `soa08Act` calculate the APV of a contract that pays 50000 m.u. at the end of the death year. The insured is 25 years old and the coverage period is 40 years.

```
data(soa08Act,package="lifecontingencies")
50000*Axn(actuarialtable=soa08Act,x=25,n=40)

## [1] 2398.544
```

Example 3: By using the table `soa08Act` calculate the APV of a contract that pays 50000 m.u. at the end of the death year. The insured is 25 years old and the coverage period is 40 years. In case that he survives 40 years, he will also receive 50000 m.u.

```
data(soa08Act,package="lifecontingencies")
50000*AExn(actuarialtable=soa08Act,x=25,n=40)

## [1] 6227.437
```

Example 4: By using the table `soa08Act` calculate the APV of an insurance that pays an increasing benefit if the insured (who is 25 years old) dies during next 2 years. If he dies during the first year, the benefit will be 1 m.u. and if he dies during the second year, the benefit will be 2 m.u.. In both cases, the benefit will be paid at the end of the death year. Do the calculation in two different ways and check that you get the same results. The technical interest rate of `soa08Act` is 6%.

```
#solution
IAxn(actuarialtable=soa08Act,x=25,n=2)

## [1] 0.003417766
```

```
#or
Axn(actuarialtable=soa08Act,x=25,n=1)+
  +2*pxt(soa08Act,25,1)*Axn(actuarialtable=soa08Act,x=26,n=1)*(1.06^(-1))

## [1] 0.003417766
```

Example 5: By using the table `soa08Act` calculate the APV of a contract that pays 1 m.u. at the end of the year of death. The insured is 25 years old and the coverage period is 1 year. Do the calculations in two different ways and check that you get the same result.

```
#solution
Axn(actuarialtable=soa08Act,x=25,n=1)

## [1] 0.001153774

qxt(soa08Act,25,1)*(1.06)^(-1) #same result

## [1] 0.001153774
```

3.1. Exercises

Exercise 1: Do the same calculation as in Example 5, in two different ways, for a coverage period of two years.

```
#Solution
Axn(actuarialtable=soa08Act,x=25,n=2)

## [1] 0.00228577

#or
qxt(soa08Act,25,1)*(1.06)^(-1)+pxt(soa08Act,25,1)*qxt(soa08Act,26,1)*(1.0
6)^(-2)

## [1] 0.00228577
```

Exercise 2: By using the table `soa08Act` calculate the APV of a contract that pays 1 m.u. if the insured (who is 25 in the moment when he underwrites the contract) survives until the retirement (65 years old). Do the calculation in two different ways and check that you get the same result.

```
#solution
Exn(soa08Act,25,40)

## [1] 0.07657786

pxt(soa08Act,25,40)*(1.06)^(-40) #same result

## [1] 0.07657786
```

Exercise 3: By using the table `soa08Act` calculate the APV of a contract that pays 1 m.u. if the insured (who is 25 when he underwrites the contract) survives until the retiremement (65 years old). If he dies before the retirement, the payment will be 1 m.u. at the end of the year when the death occurs. Do the calculation in two different ways and check that you get the same results.

```
#solution
AExn(soa08Act,25,40)

## [1] 0.1245487

Exn(soa08Act,25,40)+Axn(soa08Act,25,40)

## [1] 0.1245487

#same result
```

Exercise 4: By using the table `soa08Act` calculate the APV of an insurance contract that pays an increasing benefit if the insured (who is 25 when he underwrite the contract) dies during next three years. If he dies during the first year, the payment will be 1 m.u., if he dies during the second year, the payment will be 2 m.u., and if he dies during the third year, 3 m.u. In all cases, the benefit will be paid at the end of the death year. Do the calculation in two different ways and check that you get the same result.

```
#solution
IAxn(actuarialtable=soa08Act,x=25,n=3)

## [1] 0.006756478

Axn(actuarialtable=soa08Act,x=25,n=1)+
  +2*pxt(soa08Act,25,1)*Axn(actuarialtable=soa08Act,x=26,n=1)*(1.06^(-1))
+
  +3*pxt(soa08Act,25,2)*Axn(actuarialtable=soa08Act,x=27,n=1)*(1.06^(-2))

## [1] 0.006756478
```

Exercise 5: By using the table `soa08Act` calculate the APV of an insurance policy that pays a decreasing benefit if the insured (who is 25 years old when he underwrites the contract) dies during the next three years. If he dies during the first year, the payment will be 3 m.u., if he dies during the second year, the payment will be 2 m.u., and if he dies during the third year, 1 m.u. In all cases, the benefit will be payd at the end of the death year. Do the calculation in two different ways and check that you get the same result.

```
#solution
DAxn(actuarialtable=soa08Act,x=25,n=3)

## [1] 0.006838219

3*Axn(actuarialtable=soa08Act,x=25,n=1)+
  +2*pxt(soa08Act,25,1)*Axn(actuarialtable=soa08Act,x=26,n=1)*(1.06^(-1))
+
  +pxt(soa08Act,25,2)*Axn(actuarialtable=soa08Act,x=27,n=1)*(1.06^(-2))
```

```
## [1] 0.006838219
```

Exercise 6: By using the table `soa08Act` calculate the APV of an insurance policy that pays an increasing benefit if the insured (who is 25 years old when he underwrites the contract) dies in next 10 years. If he dies in the first year, the company will pay 100000 m.u. From that moment on, the benefit will be increased by 5000 m.u. per year (this means that if he dies during the second year the benefit will be 105000 m.u., in the third year, 110000 m.u., and so on).

```
#solution
95000*Axn(actuarialtable=soa08Act,x=25,n=10)+
5000*IAxn(actuarialtable=soa08Act,x=25,n=10)

## [1] 1325.207
```

4. Calculating the APV with lifecontingencies (two-lives)

In the last section, we will see how to calculate the APV of life insurance contracts, where the payment of the benefit depends on the survival of a group of two individuals.

4.1. Basic function for calculating the APV (two lives)

In the next table there are some basic actuarial functions for calculating the APV available in `lifecontingencies`. In all cases, the function returns the APV of the corresponding insurance contract.

Función	Descripción
Axyn	Two-lives life insurance contract (status = joint for non joint-life case and status = last for last-survivor case)
Axyzn	Life insurance contract for more that two lives (status = joint for non joint-life case and status = last last-survivor case)

Some examples of these functions:

Example 1. Calculate the APV of a life insurance contract underwritten by a couple (man aged 60 and woman aged 50) that pays a benefit when the first death occurs during the next year. The benefit is 1 m.u., at it will be paid (in the non joint life case), at the end of the year. The interest rate is 1.5%. Do the calculation in two different ways.

```
Axyzn(US07List,x=c(60,50),i=0.015, n=1, status="joint")

## [1] 0.01441865

qxyzt(US07List,x=c(60,50),t=1,status="joint")*(1.015)^(-1) # same result
```

```
## [1] 0.01441865
```

Example 2. Calculate the APV of a life insurance contract underwritten by a couple (man aged 60 and woman aged 50) that pays a benefit when the first death occurs during the next year. The benefit is 50000 m.u., at it will be payd, in the non joint life case, at the end of the year. The interest rate is 1.5%. Do the calculation in two different ways.

```
50000*Axyzn(US07List,x=c(60,50),i=0.015, n=1, status="joint")

## [1] 720.9323

50000*qxyzt(US07List,x=c(60,50),t=1,status="joint")*(1.015)^(-1) # same r
esult

## [1] 720.9323
```

Example 3. Calculate the APV of a life insurance that pays a benefit when all member of a couple (man aged 60 and woman aged 50) die during the next year. The benefit is 50000 m.u., at it will be payd, in the last-survivor case, at the end of the year. The interest rate is 1.5%. Do the calculation in two different ways.

```
50000*Axyzn(US07List,x=c(60,50),i=0.015, n=1, status="last")

## [1] 1.830881

50000*qxyzt(US07List,x=c(60,50),t=1,status="last")*(1.015)^(-1) # same re
sult

## [1] 1.830881
```

4.1. Exercises

Exercise 1: Calculate the APV of a life insurance that pays a benefit when the first death (non joint life case) occurs in a couple (man aged 60 and woman aged 50) during the next two years. The benefit is 1 m.u., at it will be payd at the end of the year when the first death occurs. The interest rate is 1.5%. Do the calculation in two different ways.

```
#solution
Axyzn(US07List,x=c(60,50),i=0.015, n=2, status="joint")

## [1] 0.0295044

qxyzt(US07List,x=c(60,50),t=1,status="joint")*(1.015)^(-1)+
  +pxyzt(US07List,x=c(60,50),t=1,
         status="joint")*qxyzt(US07List,x=c(61,51),t=1,
                                            status="joint")*(1.015)^(-2) # same result

## [1] 0.0295044
```

Exercise 2: Calculate the APV of a life insurance that pays a benefit when the last death occurs (last-survivor case) of a couple (man aged 60 and woman aged 50) during the next two years. The benefit is 1 m.u., at it will be payd at the end of the year when the last death occurs. The interest rate is 1.5%. Do the calculation in two different ways.

```
#solution
Axyzn(US07List,x=c(60,50),i=0.015, n=2, status="last")

## [1] 0.0001551046

#otra manera de hacer el mismo cálculo
qxyzt(US07List,x=c(60,50),t=1,status="last")*(1.015)^(-1)+
  +(pxyzt(US07List,x=c(60,50),t=1,
              status="last")-pxyzt(US07List,x=c(60,50),t=2,status="last"))
*(1.015)^(-2)

## [1] 0.0001551046
```

Exercise 3: By using the life table `US07List` calculate the APV of the following contracts:

a) Life insurance contract that pays a benefit when the first death occurs, underwritten by a couple (man aged 48 and woman aged 45) , the covarage period is 5 years. The benefit is 15000 euros and it is payd at the end of the year when the first death occurs. Interest rate: 1%.

b) Life insurance contract that pays a benefit when the last death occurs (last-survivor case), underwritten by a couple (man aged 57 and woman aged 55), the coverage period is 10 years. The benefit is 180000 euros and it is payd at the end of the year when the last death occurs. The interest rate is 1%.

```
#a)
15000*Axyzn(US07List,x=c(48,45),i=0.01,n=5,status="joint")

## [1] 579.8566

#b)
180000*Axyzn(US07List,x=c(57,55),i=0.01,n=10,status="last")

## [1] 1379.742
```

References

Ayuso, M., Corrales, H., Guillén, M., Perez-Marin, A. M. and Rojo, J. L. (2007) Estadística Actuarial Vida. Ediciones de la Universidad de Barcelona. Barcelona.

Spedicato, G. A. (2013) “The lifecontingencies package: Performing financial and actuarial mathematics calculations in R”, Journal of Statistical Software, 55, 1-36.