Intro to the lifecontingencies R package

Giorgio Alfredo Spedicato, Ph.D C.Stat ACAS

19 settembre, 2018

Intro

- The lifecontingencies package (Spedicato 2013) will be introduced.
- As first half 2017 it is the first R (Team 2012) package merging demographic and financial mathematics function in order to perform actuarial evaluation of life contingent insurances (annuities, life insurances, endowments, etc).
- The applied examples will shown: how to load the R package, how to perform basic financial mathematics and demographic calculations, how to price and reserve financial products.

- The final example will show how to mix lifecontingencies and demography (Rob J Hyndman et al. 2011) function to assess the mortality development impact on annuities.
- The interested readers are suggested to look to the package's vignettes (also appeared in the Journal of Statistical Sofware) for a broader overview. (Dickson, Hardy, and Waters 2009; and Mazzoleni 2000) provide and introduction of Actuarial Mathematics theory.
- Also (Charpentier 2012) and (Charpentier 2014) discuss the software.

Loading the package

- The package is loaded using

```
library(lifecontingencies) #load the package
```

- It requires a recent version of R (>=3.0) and the markovchain package (Spedicato, Giorgio Alfredo 2015). The development version of the package requires also Rcpp package (Eddelbuettel 2013).

Package's Financial Mathematics Functions

Interest functions

- interest2Discount, discount2Interest: from interest to discount and reverse;
- interest2Intensity, intensity2Interest: from intensity to interest and reverse;
- convertible2Effective, effective2Convertible: from convertible interest rate to effective one.

- $(1+i) = \left(1 + \frac{i^{(m)}}{m}\right)^{m} = e^{\delta}$
- $e^{\delta} = \left(1 - \frac{d^{(m)}}{m}\right)^{-m} = (1-d)^{-1}$

```r
#interest and discount rates
interest2Discount(i=0.03)
```

```
## [1] 0.02912621
```

```r
discount2Interest(interest2Discount(i=0.03))
```

```
## [1] 0.03
```

```r
#convertible and effective interest rates
convertible2Effective(i=0.10,k=4)
```

```
## [1] 0.1038129
```

Annuities and future values

- annuity: present value (PV) of an annuity;
- accumulatedValue: future value of constant cash flows;
- decreasingAnnuity, increasingAnnuity: increasing and decreasing annuities.

- $a_{\overline{n}|} = \frac{1-(1+i)^{-n}}{i}$
- $s_{\overline{n}|} = a_{\overline{n}|} * (1+i)^n = \frac{(1+i)^n - 1}{i}$
- $\ddot{a}_{\overline{n}|} = a_{\overline{n}|} * (1+i)$

```r
annuity(i=0.05,n=5) #due
```

```
## [1] 4.329477
```

```r
annuity(i=0.05,n=5,m=1) #immediate
```

```
## [1] 4.123311
```

```r
annuity(i=0.05,n=5,k=12) #due, with
```

```
## [1] 4.42782
```

```r
# fractional payemnts
```

- $\frac{\ddot{a}_{\overline{n}|} - n * v^n}{i} = (Ia)$
- $\frac{n - a_{\overline{n}|}}{i} = (Da)$
- $(Ia_n) + (Da_n) = (n + 1) * a_{\overline{n}|}$

```r
irate=0.04; term=10
increasingAnnuity(i=irate,n=term)+decreasingAnnuity(i=irate,
n=term)-(term+1)*annuity(i=irate,n=term)
```

```
## [1] -1.421085e-14
```

Other functions

- presentValue: PV of possible varying CFs.
- duration, convexity: calculate duration and convexity of any stream of CFs.

- $PV = \sum_{t \in T} CF_t * (1 + i_t)^{-t}$
- $D = \frac{1}{P(0)} \sum_{t=\tau}^{T} t \frac{c_t}{(1+r)^t}$
- $C = \frac{1}{P(1+r)^2} \sum_{i=1}^{n} \frac{t_i(t_i+1)F_i}{(1+r)^{t_i}}$

```r
#bond analysis
irate=0.03
cfs=c(10,10,10,100)
times=1:4
#compute PV, Duration and Convexity
presentValue(cashFlows = cfs,
timeIds = times,
interestRates = irate)
```

```
## [1] 117.1348
```

```r
duration(cashFlows = cfs,
timeIds = times, i = irate)
```

```
## [1] 3.512275
```

```r
convexity(cashFlows = cfs,
timeIds = times, i = irate)
```

Intro

- Lifecontingencies offers a wide set of functions for demographic analysis;
- Survival and death probabilities, expected residual lifetimes and other function can be easily modeled with the R package;
- Creation and manipulation of life table is easy as well.

Package's demographic functions

Table creation and manipulation

- new lifetable or actuarialtable methods.
- print, plot show methods.
- probs2lifetable function to create tables from probabilities

- $\{l_0, l_1, l_2, \ldots, l_\omega\}$
- $L_x = \frac{l_x + l_{x+1}}{2}$
- $q_{x,t} = \frac{d_{x+t}}{l_x}$

```r
data("demoIta")
sim92<-new("lifetable",x=demoIta$X,
           lx=demoIta$SIM92, name='SIM92')
getOmega(sim92)
```

```
## [1] 108
```

```r
tail(sim92)
```

```
##       x lx
## 104 103 47
## 105 104 24
## 106 105 11
## 107 106  5
## 108 107  2
## 109 108  1
```

Life tables' functions

- dxt, deaths between age x and x + t, tdx
- pxt, survival probability between age x and x + t, tpx
- pxyzt, survival probability for two (or more) lives, tpxy
- qxt, death probability between age x and x + t, tqx
- qxyzt, death probability for two (or more) lives, tqxy
- Txt, number of person-years lived after exact age x, tTx
- mxt, central death rate, tmx
- exn, expected lifetime between age x and age x + n, nex
- exyz, n-year curtate lifetime of the joint-life status

- $p_{x,t} = 1 - q_{x,t} = \frac{l_{x+t}}{l_x}$
- $e_{x,n} = \sum_{t=1}^{n} p_{x,t}$

```r
#two years death rate
qxt(sim92, x=65,2)
```

```
## [1] 0.04570079
```

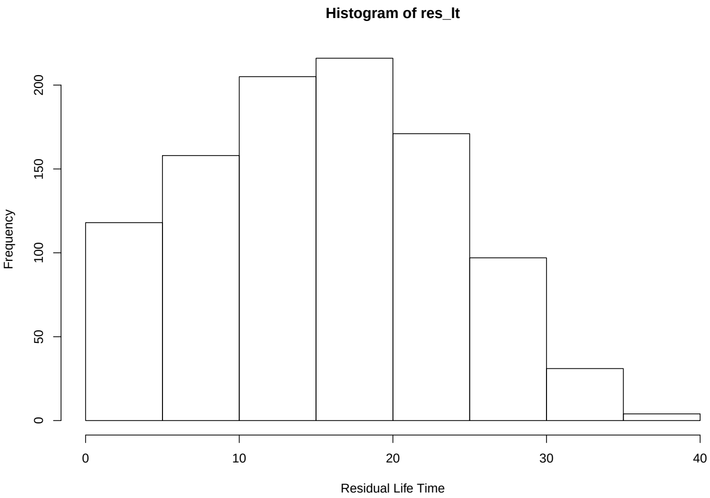
```r
#expected residual lifetime between x and x+n
exn(sim92, x=25,n = 40)
```

```
## [1] 37.86183
```

Simulation

- rLife, sample from the time until death distribution underlying a life table
- rLifexyz, sample from the time until death distribution underlying two or more lives

```
#simulate 1000 samples of residual life time
res_lt<-rLife(n=1000,object = sim92,x=65)
hist(res_lt,xlab="Residual Life Time")
```



Assessing longevity impact on annuities using lifecontingencies and demography

- This part of the presentation will make use of the demography package to calibrate Lee Carter (Lee and Carter 1992) model, $log\left(\mu_{x,t}\right) = a_x + b_x * k_t \rightarrow p_{x,t} = exp^{-\mu_{x,t}}$,projecting mortality and implicit life tables.

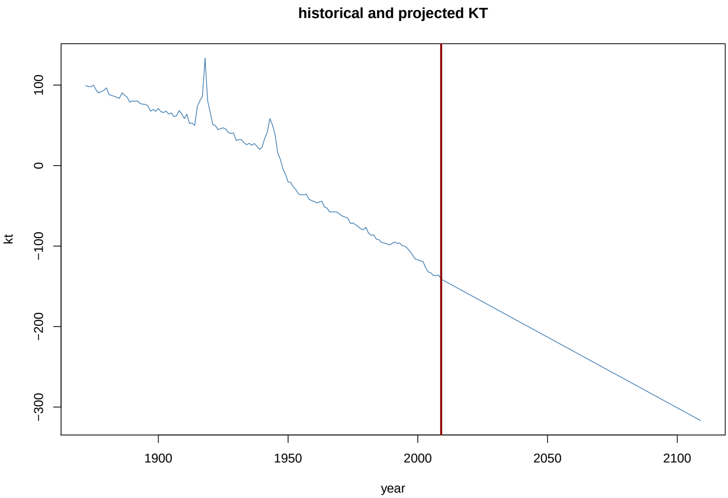
```r
#load the package and the italian tables
library(demography)
#italyDemo<-hmd.mx("ITA", username="yourUN",
#password="yourPW")
load(file="mortalityDatasets.RData") #load the dataset
```

- Lee Carter model is calibrated using lca function of demography package.
- Then an arima model is used to project (extrapolate) the underlying k_t over the historical period.

```
#calibrate lee carter
italy.leecarter<-lca(data=italyDemo,series="total",
                     max.age=103,adjust = "none")
#perform modeling of kt series
kt.model<-auto.arima(italy.leecarter$kt)
#projecting the kt
kt.forecast<-forecast(kt.model,h=100)
```

-The code below generates the matrix of prospective life tables

```
#indexing the kt
kt.full<-ts(union(italy.leecarter$kt, kt.forecast$mean),
            start=1872)
#getting and defining the life tables matrix
mortalityTable<-exp(italy.leecarter$ax
+italy.leecarter$bx%*%t(kt.full))
rownames(mortalityTable)<-seq(from=0, to=103)
colnames(mortalityTable)<-seq(from=1872,
to=1872+dim(mortalityTable)[2]-1)
```


historical and projected KT
kt
100
0
−100
−200
−300
1900
1950
2000
2050
2100
year

- now we need a function that returns the one-year death probabilities given a year of birth (cohort.

```
getCohortQx<-function(yearOfBirth)
{
  colIndex<-which(colnames(mortalityTable)
                  ==yearOfBirth) #identify
  #the column corresponding to the cohort
  #definex the probabilities from which
  #the projection is to be taken
  maxLength<-min(nrow(mortalityTable)-1,
                 ncol(mortalityTable)-colIndex)
  qxOut<-numeric(maxLength+1)
  for(i in 0:maxLength)
    qxOut[i+1]<-mortalityTable[i+1,colIndex+i]
  #fix: we add a fictional omega age where
  #death probability = 1
  qxOut<-c(qxOut,1)
  return(qxOut)
```

- Now we use such function to obtain prospective life tables and to perform actuarial calculations. For example, we can compute the APV of an annuity on a workers' retiring at 65 assuming he were born in 1920, in 1950 and in 1980. We will use the interest rate of 1.5% (the one used to compute Italian Social Security annuity factors).
- The first step is to generate the life and actuarial tables

```
#generate the life tables
qx1920<-getCohortQx(yearOfBirth = 1920)
lt1920<-probs2lifetable(probs=qx1920,type="qx",
name="Table 1920")
at1920<-new("actuarialtable",x=lt1920@x,
lx=lt1920@lx,interest=0.015)
qx1950<-getCohortQx(yearOfBirth = 1950)
lt1950<-probs2lifetable(probs=qx1950,
type="qx",name="Table 1950")
at1950<-new("actuarialtable",x=lt1950@x,
lx=lt1950@lx,interest=0.015)
qx1980<-getCohortQx(yearOfBirth = 1980)
```

- Now we can evaluate $\ddot{a}_{65}$ and $\mathring{e}_{65}$ for workers born in 1920, 1950 and 1980 respectively.

```
cat("Results for 1920 cohort","\n")
```

```
## Results for 1920 cohort
```

```
c(exn(at1920,x=65),axn(at1920,x=65))
```

```
## [1] 16.51391 15.14127
```

```
cat("Results for 1950 cohort","\n")
```

```
## Results for 1950 cohort
```

```
c(exn(at1950,x=65),axn(at1950,x=65))
```

```
## [1] 18.72669 16.83391
```

Intro on Actuarial Mathematics Funcs

- The lifecontingencies package allows to compute all classical life contingent insurances.
- Stochastic calculation varying expected lifetimes are possible as well.
- This makes the lifecontingencies package a nice tool to perform actuarial computation at command line on life insurance tasks.

Creating actuarial tables

- Actuarial tables are stored as S4 object.
- The l_x, x, interest rate and a name are required.
- The print method return a classical actuarial table (commutation functions)

```r
data("demoIta")
sim92Act<-new("actuarialtable",x=demoIta$X,
lx=demoIta$SIM92, name='SIM92')
sif92Act<-new("actuarialtable",x=demoIta$X,
lx=demoIta$SIF92, name='SIF92')
head(sim92Act)
```

```
##   x     lx
## 1 0 100000
## 2 1  99121
## 3 2  99076
## 4 3  99043
## 5 4  99018
## 6 5  98997
```

Life insurances functions

Classical life contingent insurances

- Exn, pure endowment: $A_{x:\overline{n|}}^{\;\;1}$
- axn, annuity: $\ddot{a}_x = \sum_{k=0}^{\omega-x} v^k * p_{x,k} = \sum_{k=0}^{\omega-x} \ddot{a}_{\overline{K+1|}} p_{x,k} q_{x+k,1}$
- Axn, life insurance: $A_{x:\overline{n|}}^1 = \sum_{k=0}^{n-1} v^{k-1} * p_{x,k} * q_{x+k,1}$
- AExn, endowment: $A_{x:\overline{n|}} = A_{x:\overline{n|}}^{\;\;1} + A_{x:\overline{n|}}^1$

```r
100000*Exn(sim92Act,x=25,n=40)
```

```
## [1] 24822.27
```

```r
100000*AExn(sim92Act,x=25,n=40)
```

```
## [1] 33104.86
```

```r
1000*12*axn(sim92Act,x=65,k=12)
```

```
## [1] 139696.3
```

```r
100000*Axn(sim92Act,x=25,n=40)
```

```
## [1] 8282.588
```

Additional life contingent insurances

- Increasing and decreasing term insurances and annuities
- $(n-1) * A^{1}_{x:\overline{n|}} = (IA)^{1}_{x:\overline{n|}} + (DA)^{1}_{x:\overline{n|}}$

```r
IAxn(sim92Act,x=40,n=10)+DAxn(sim92Act,x=40,n=10)
```

```
## [1] 0.2505473
```

```r
(10+1)*Axn(sim92Act,x=40,n=10)
```

```
## [1] 0.2505473
```

Insurances on multiple lifes

- First survival and last survival status, both for insurances and annuities
- $A_{xy}+A_{\overline{xy}}=A_x+A_y$
- $a_{xy}+a_{\overline{xy}}=a_x+a_y$

```
fr_pay=12
1000*fr_pay*axyzn(tablesList = list(sim92Act,sif92Act),
x = c(64,67),status="last",k=fr_pay)
```

```
## [1] 185101.3
```

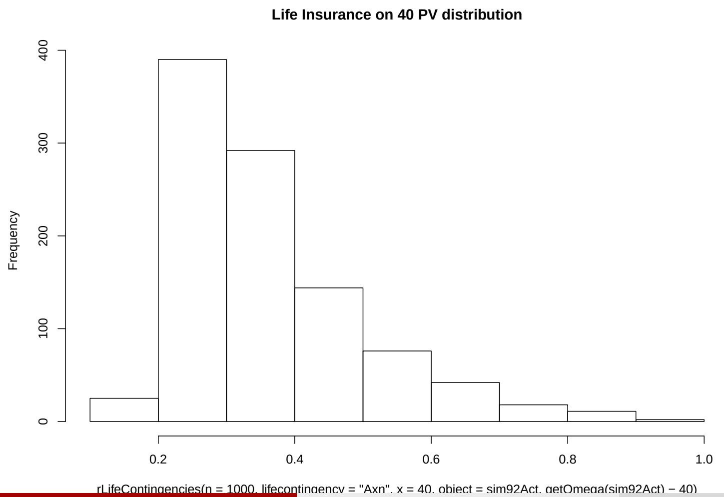
```
1000*fr_pay*(axn(sim92Act,x=64,k=fr_pay)+axn(sif92Act,
x=67,k=fr_pay)-axyzn(tablesList = list(sim92Act,sif92Act),
x = c(64,67),status="joint",k=fr_pay))
```

```
## [1] 185101.3
```

Simulation

- It is possible to simulate the PV of insured benefit distributions.
- The rLifeContingencies function is used for single life benefit insurance.
- The rLifeContingenciesXyz function is used for multiple lifes benefits.

```r
hist(rLifeContingencies(n = 1000,lifecontingency = "Axn",
x = 40,object = sim92Act,getOmega(sim92Act)-40),
main="Life Insurance on 40 PV distribution")
```



Bibliography I

Bibliography II

Charpentier, Arthur. 2012. "Actuarial Science with R 2: Life Insurance and Mortality Tables." http://freakonometrics.blog.free.fr/index.php?post/2012/04/04/Life-insurance,-with-R,-Meielisalp.

———. 2014. *Computational Actuarial Science*. The R Series. Cambridge University Press.

Dickson, D.C.M., M.R. Hardy, and H.R. Waters. 2009. *Actuarial Mathematics for Life Contingent Risks*. International Series on Actuarial Science. Cambridge University Press.

Eddelbuettel, Dirk. 2013. *Seamless R and C++ Integration with Rcpp*. New York: Springer.

Lee, R.D., and L.R. Carter. 1992. "Modeling and Forecasting U.S. Mortality." *Journal of the American Statistical Association* 87 (419): 659–75. doi:10.2307/2290201.

Mazzoleni, P. 2000. *Appunti Di Matematica Attuariale*. EDUCatt.