Assignment 1

Cosmo Strozza

I was already registered at www.mortality.org. I used the following function, that is included in "HMDHFDplus" library, to download single years, single period death counts and exposure of Italy.

```
library(HMDHFDplus)

italy_death <- readHMDweb(CNTRY = "ITA", item = "Deaths_1x1",
                          username = "***", password = "***")
italy_exp <- readHMDweb(CNTRY = "ITA", item = "Exposures_1x1",
                        username = "***", password = "***")
```

In order to create two matrices from the two data frame downloaded from www.mortality.org, I defined two vectors: years and ages with duplicate elements removed. I used these two vectors to identify rows and columns of D and E that are respectively the matrix of deaths and exposures with counts for each year and age for the whole population.

```
x <- c(0,1, seq(2,110,1))
years <- unique(italy_death$Year)
ages <- unique(italy_death$Age)

D <- matrix(italy_death$Total, nrow = length(ages), dimnames = list(ages,
years))
E <- matrix(italy_exp$Total, nrow = length(ages), dimnames = list(ages,
years))
```

I used the following function to calculate life expectancy at birth and the nominator of coefficient of variation of the life distribution of death. This function needs as input a vector of years and two matrices about deaths and exposures and it gives as output a list. This list is composed by a data frame (complete life table with life expectancy and the numerator of the coefficient of variation) and a summary with life expectancy at birth and the final expression of the coefficient of variation.

```
LTFun <- function (x, Dx, Nx) {

  nx <- c(diff(x), Inf)
  nmx <- Dx / Nx
  npx <- exp(-nx*nmx)
  nqx <- 1-npx
  lx <- cumprod(c(1, npx[-length(npx)]))
  ndx <- c(diff(-lx), lx[length(lx)])
  nLx <- -nx*ndx/log(npx); nLx[is.nan(nLx)] <- 0
  Tx <- rev(cumsum(rev(nLx)))
  ex <- Tx/lx
```

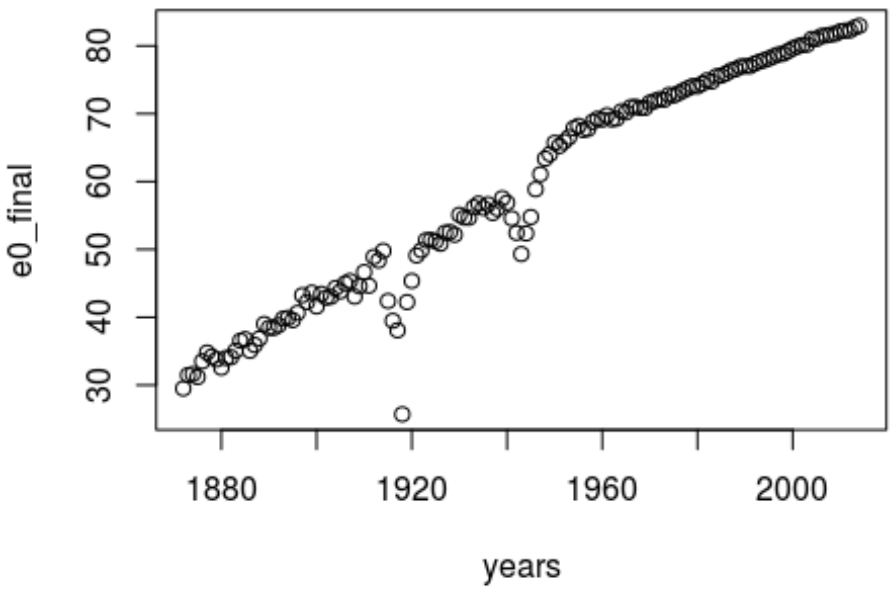
```
  cv <- sqrt(cumsum(ndx*(x+0.5-ex)^2))

  list(
    lt = data.frame(
      x, nx, nmx, npx, nqx, lx, ndx, nLx, Tx, ex, cv
    ),
    summary = c(e0 = ex[[1]], m = x[which.max(ndx)], ltdr = 1/ex[[1]], cv0 =
cv[[1]]/ex[[1]])
  )
}
```

Before to use the function I created an empty list called e0t where to insert the result of the function with deaths and exposures of Italy. In order to fill the list I used a for loop as long as the number of the columns of D (or E is equivalent). Then I saved the life expectancies at birth and the coefficient of variation, both per year, in two vectors.

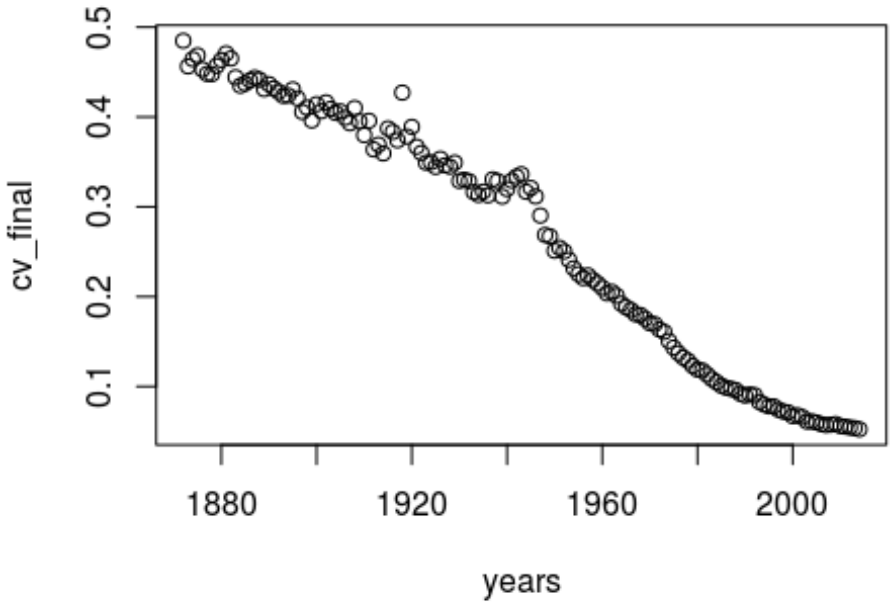
```
e0t <- vector(mode = 'list', length = ncol(D))

for (i in 1:ncol(D)) {
  e0t[[i]] <- LTFun(x = ages, Dx = D[,i], Nx = E[,i])
}

e0_final <- sapply(e0t, function(x) {x$summary['e0']})
cv_final <- sapply(e0t, function(x) {x$summary['cv0']})
```

Shown below there are some plot of e0, cv and e0 vs cv. As we can see in the last graphic, it seems to be a linear relationship between life expectancy at birth and coefficient of variation. As the e0 increases, the cv decreses.

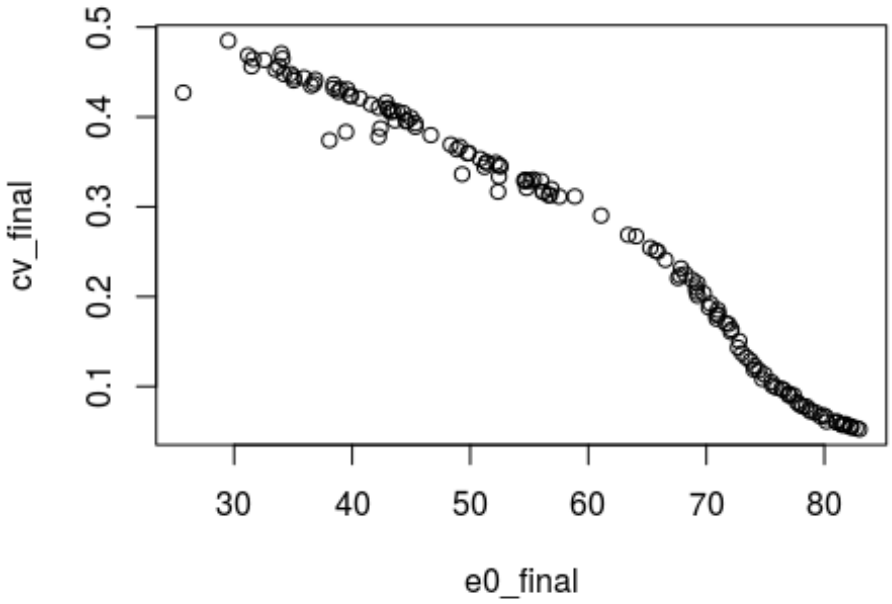
```
plot(years,e0_final)
```




```
plot(years,cv_final)
```




```
plot(e0_final,cv_final)
```



As a proof I first calculated the correlation between the two variables and then i fitted a model using cv as dependent variable and e0 as independent variable. The results are shown below.

```
cor(e0_final,cv_final)

## [1] -0.983238

model <- lm(cv_final ~ e0_final)
summary(model)

##
## Call:
## lm(formula = cv_final ~ e0_final)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.116608 -0.021382  0.001787  0.019462  0.044295
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.7578109  0.0079202   95.68   <2e-16 ***
## e0_final    -0.0083350  0.0001302  -64.03   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02569 on 141 degrees of freedom
```

```
## Multiple R-squared:  0.9668, Adjusted R-squared:  0.9665
## F-statistic:  4100 on 1 and 141 DF,  p-value: < 2.2e-16

plot(e0_final,cv_final)
abline(lm(cv_final ~ e0_final), col='red')
```