# Local Poisson regression

Biel Caballero Vergés, Svenja Menzenbach and Kleber Enrique Reyes Illescas

2023-11-20

## 1. Bandwidth choice for the local Poisson regression

```
h.cv.sm.poisson <- function(x, y, rg.h = NULL, l.h = 20) {
  cv.h <- numeric(l.h)

  if (is.null(rg.h)) {
    hh <- c(h.select(x, y, method = "cv"), h.select(x, y, method = "aicc"))
    rg.h <- range(hh) * c(0.5, 1.5)
  }

  gr.h <- exp(seq(log(rg.h[1]), log(rg.h[2]), length.out = l.h))

  for (i in 1:length(gr.h)) {
    cv.h[i] <- loglik.CV(x, y, gr.h[i])
  }

  return(list(h = gr.h, cv.h = cv.h, h.cv = gr.h[which.min(cv.h)]))
}
```

```
loglik.CV <- function(x, y, h) {
    n <- length(x)

    lambda <- sapply(1:n,
                 function(i, x, y, h) {
        sm.poisson(x = x[-i], y = y[-i], h = h, eval.points = x[i], display = "none")$estimate
    }, x, y, h)

    # return(-sum( log(exp(-lambda)*((lambda^y)/factorial(y))))/n )
    return(-sum(-lambda + log(lambda^y/factorial(y)))/n )
}
```

## 2. Local Poisson regression for Country Development Data

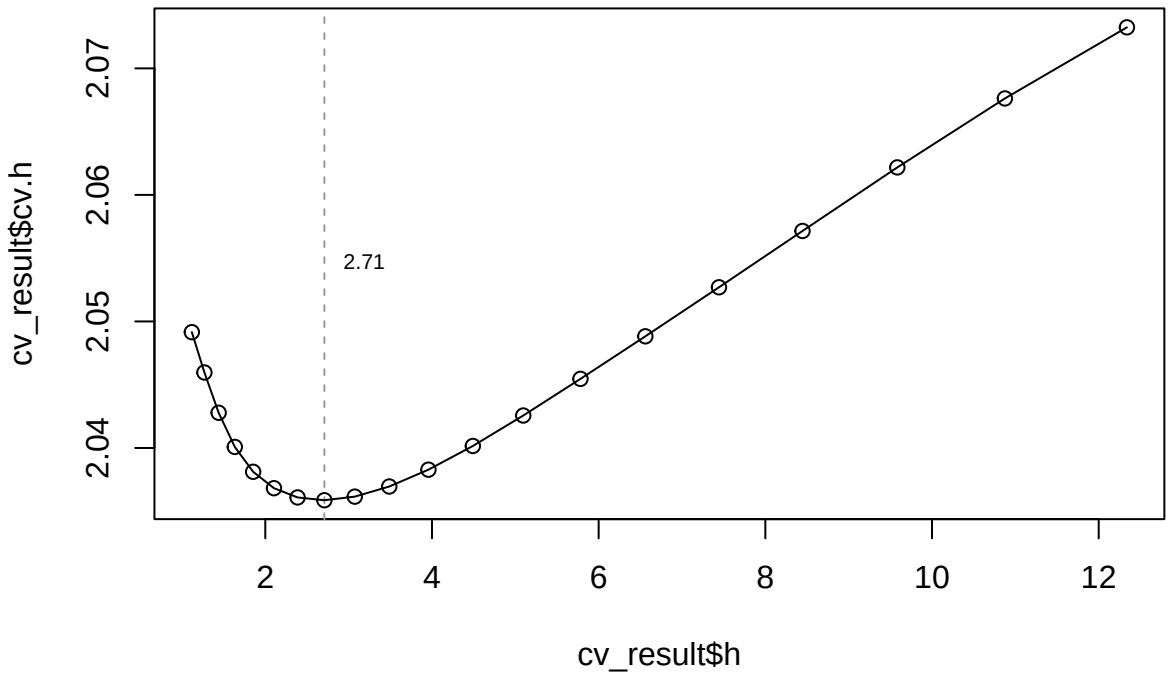
```
# Load required library
library(sm)
```

```
## Package 'sm', version 2.2-5.7: type help(sm) for summary information
```

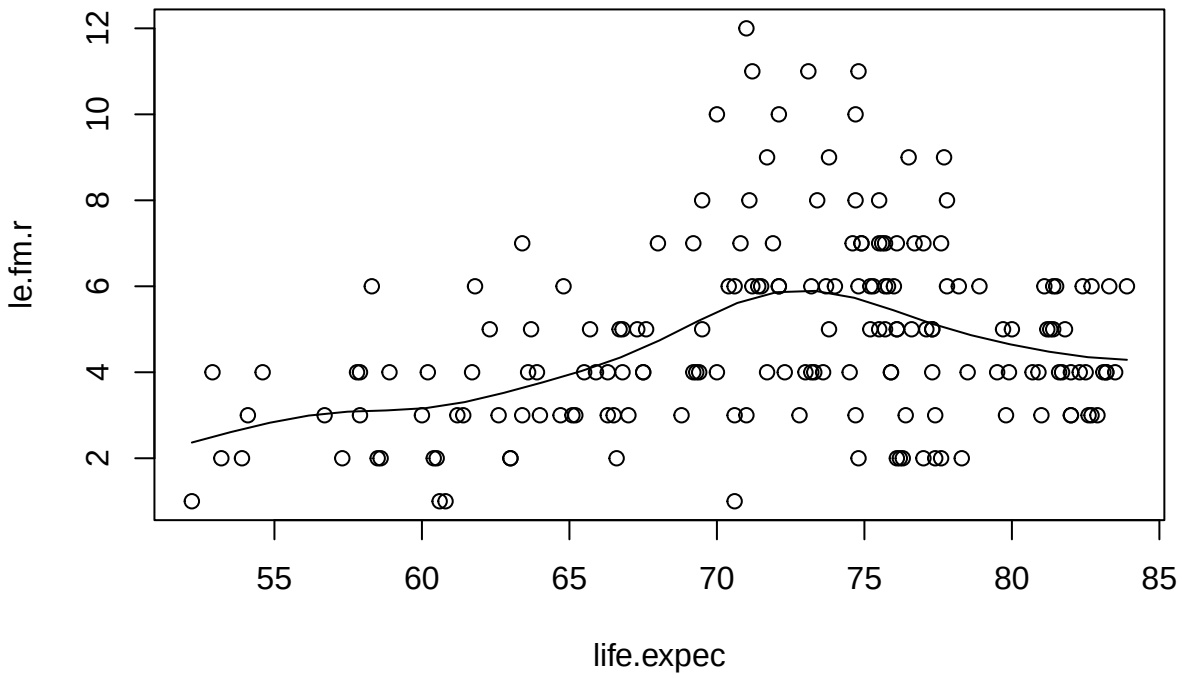
```r
countries <- read.csv2(file="HDI.2017.subset.csv",row.names = 1)

life.expec <- countries$Life.expec
le.fm.r <- round(countries$le.fm)

cv_result <- h.cv.sm.poisson(life.expec, le.fm.r)
plot(cv_result$h, cv_result$cv.h)
selected.bandwidth <- cv_result$h.cv
abline(v = selected.bandwidth, col="8", lty=2)
mid.point <- (max(cv_result$cv.h)+min(cv_result$cv.h))/2
text(selected.bandwidth, mid.point, round(selected.bandwidth*100)/100,cex=0.65, pos=4,col=1)
lines(cv_result$h, cv_result$cv.h)
```




```r
model <- sm.poisson(x = life.expec, y = le.fm.r, h = selected.bandwidth, col=1)
```




```
summary(model)
```

```
##                  Length Class  Mode
## call              5     -none- call
## eval.points      25     -none- numeric
## estimate         25     -none- numeric
## lower            25     -none- numeric
## upper            25     -none- numeric
## linear.predictor 25     -none- numeric
## se               25     -none- numeric
## deviance         25     -none- numeric
## data              4     -none- list
```