## Non-parametric estimation of a density: Cross Validation

*Data: geyser.* Waiting time between eruptions and the duration of the eruption for the Old Faithful geyser in Yellowstone National Park, Wyoming, USA.

```
library(MASS)
data(geyser)
str(geyser)
'data.frame':   299 obs. of  2 variables:
 $ waiting : num  80 71 57 80 75 77 60 86 77 56 ...
 $ duration: num  4.02 2.15 4 4 4 ...
help(geyser)
```

We are interested by the estimation of the density of the variable Waiting (Waiting time to next eruption).

1. Why attach ? why density ? what is the value of the bandwidth ? what is the kernel $K$?

   ```
   attach(geyser)
   Kernel_density<-density(waiting)
   plot(Kernel_density)
   ```

2. The task now is to use other kernel $K$ in order to observe their influence in the density estimation.

   ```
   par(mfrow=c(1,4))
   plot(density(waiting, kernel="gaussian"), main="Gaussian Kernel")
   plot(density(waiting, kernel="epanechnikov"), main="Epanechnikov Kernel")
   plot(density(waiting, kernel="rectangular"), main="Rectangular Kernel")
   plot(density(waiting, kernel="triangular"), main="Triangular Kernel")
   ```

   Conclusion ?

3. The purpose now is to observe the influence of the bandwidth $h$ in the estimation of the density.

   ```
   par(mfrow=c(2,4))
   plot(density(waiting,bw=0.5), main="Bandwidth 0.5")
   plot(density(waiting,bw=1), main="Bandwidth 1")
   plot(density(waiting,bw=2), main="Bandwidth 2")
   plot(density(waiting,bw=3), main="Bandwidth 3")
   plot(density(waiting,bw=4), main="Bandwidth 4")
   plot(density(waiting,bw=6), main="Bandwidth 6")
   plot(density(waiting,bw=8), main="Bandwidth 8")
   plot(density(waiting,bw=12), main="Bandwidth 12")
   ```

4. **Cross-Validation.** The purpose here is to construct the optimal bandwidth (for a fixed kernel: the Gaussian kernel) in the sense of the MISE. Recall that

$$MISE(h) = E\left(\int (f_h^K)^2 - 2\int f_h^K f\right) + \text{a constant not depending on h}$$

and its estimator up to some constant (non depending on $h$) is

$$\|\hat{f}_h^K\|_2^2 - \frac{2}{(n-1)nh}\sum_{i=1}^{n}\sum_{j\neq i} K(\frac{X_i - X_j}{h})$$

which can be approximated by

$$\tilde{J}_h = \frac{1}{V}\sum_{v=1}^{V}\left(\int (\hat{f}_{n,h}^v(x))^2 dx - \frac{2}{|C_v|(|C_v|-1)h}\sum_{i,j\in C_v,\ i\neq j} K(\frac{X_i - X_j}{h})\right)$$

where $\hat{f}_{n,h}^v$ is the estimator of calculated by using all the data except that on the set $C_v$.

```
# Global Error
 total_risk <- c()
# We use a grid for h with length  1/50 on the interval [0,8]
 for (i in 1:400)
 {
 # Construction of the risk vector
 paq_risk <- c()
 h <- 8*i/400
 # The 4 first sets are considered on the same loop
 for (j in 1:4)
 {
 # We construct the estimator over all the data except those on the jth set
 data <- waiting[-((60*(j-1)+1):(60*j))]
 estim <- density(data,bw=h)
 # We approximate the integral of the square of the estimator
 long <- length(estim$y)
 int_square <- mean( (estim$x[2:long]-estim$x[1:(long-1)]) * (estim$y[2:long]^2))
 # We approximate the second part of the risk using the data of the set j
 data <- waiting[(60*(j-1)+1):(60*j)]
 estim <- density(data,bw=h)
 nb_obs <- length(data)
 # We create a vectorial empty variable,
 terms <- c()
 # for each observation of the set v
 for (k in 1:(nb_obs))
 {
```

```
 index <- which.min(abs(estim$x-data[k]))
 terms <- c(terms,estim$y[index]-(2*pi)^{-1/2}/nb_obs/h)
 }
second_term <- 2/(nb_obs-1)*sum(terms)
 paq_risk <- c(paq_risk,int_square-second_term)
 }
 # We add the last set of data
 data <- waiting[-(241:299)]
 estim <- density(data,bw=h)
 long <- length(estim$y)
 int_square <- mean( (estim$x[2:long]-estim$x[1:(long-1)]) * (estim$y[2:long]^2))
 data <- waiting[241:299]
 estim <- density(data,bw=h)
 nb_obs <- length(data)
 terms <- c()
 for (k in 1:(nb_obs))
 {
 index <- which.min(abs(estim$x-data[k]))
 terms <- c(terms,estim$y[index]-(2*pi)^{-1/2}/nb_obs/h)
 }
 second_term <- 2/(nb_obs-1)*sum(terms)
 paq_risk <- c(paq_risk,int_square-second_term)
 total_risk <- c(total_risk,mean(paq_risk))
 }
# We extract the value of h for which the estimated risk is minimal
 h_CV <- 8*which.min(total_risk)/400
 h_CV
 Final_estim <- density(waiting,bw=h_CV)
 plot(Final_estim)
```

## Explanation of the instructions.

1. In practice, we use a grid with length $8 \times 50 = 400$ on $h \in [0, 8]$ and cut the sample with length $n = 299$ variables into 4 of packets of length 60 and a fifth package of size 59. We use

```
 # We use a grid for h with step 1/50 on [0,8]
for (i in 1:400)
 {
h <- 8*i/400
# the 4 first blocks are considered together
for (j in 1:4)
 {
```

```
 # we construct the kernel estimator based on all the observation
 #only on that of the jth block
data <- waiting[-((60*(j-1)+1):(60*j))]
 estim <- density(data,bw=h)
 # we approximate the first term of the MISE
 # we approximate the second term of the MISE
 # we obtained an approximated value of the MISE
}
 # We add the 5th block
 data <- waiting[-(241:299)]
 estim <- density(data,bw=h)
 # We estimate the risk on this block
 # We average all risks
# We store the estimated  risk  for the window
 }
 # The value of h for which the estimated risk is minimal is extracted
```

2. Step 2. The second step is to approximate the integral $\int(\hat{f}^v_{n,h}(x))^2dx$ by a finite riemann sum of the form

$$\frac{1}{M}\sum_{v=1}^{M-1}(x_{i+1}-x_i)[\hat{f}^v_{n,h}(x_{i+1})]^2$$

where $(x_i)_{1\leq i\leq M}$ is a grid on the horizontal axis. The output density is a list including the first variable x is a set of points on the horizontal axis which forms a gate and the second variable is the value of the estimator at these points (vector of the same length).

We then use

```
# We approximate the integral of the square of the estimator
long <- length(estim$y)
int_square <- mean( (estim$x[2:long]-estim$x[1:(long-1)]) * (estim$y[2:long]^2))
```

3. For the second part of the estimation of the risk on the $v$-th block for the bandwidth $h$ we remark that,

$$\begin{aligned}
&\frac{2}{|C_v|(|C_v|-1)h}\sum_{i,j\in C_v,\, i\neq j}K(\frac{X_i-X_j}{h})\\
&=\frac{2}{(|C_v|-1)}\sum_{i\in C_v}\left\{\frac{1}{|C_v|h}\sum_{j\in C_v}K(\frac{X_i-X_j}{h})-\frac{K(0)}{|C_v|h}\right\}\\
&=\frac{2}{(|C_v|-1)}\sum_{i\in C_v}\left\{f^{C_v}_{n,h}(X_i)-\frac{K(0)}{|C_v|h}\right\}
\end{aligned}$$

where $f^{C_v}_{n,h}(X_i)$ is the kernel estimator with bandwidth $h$ constructed using the observations on the block $C_v$ and taken at the observation $X_i$ and $K(0)=(2\pi)^{-1/2}$ The density function lets not directly assess the kernel estimator at one point we will have find on the grid abscissa the nearest point of our observation $X_i$ and use the value of the corresponding ordinated. We use then,

```
# we approximate the second part of the risk using data of the j-th block
data <- waiting[(60*(j-1)+1):(60*j)]
estim <- density(data,bw=h)
nb_obs <- length(data)
# we construct an empty vectorial variable
terms <- c()
# for each observation of the block v
For (k in 1:nb_obs)
{
# We search from the abscissa of which is estimated the nearest
#of the observation value X
index <- which.min(abs(estim$x-donnees[k]))
# The value of the kernel estimator is selected at this point and the previous
#concatenates We do not forget to remove the value K(0)/|C_v|/h
terms <- c(terms,estim$y[indice]-(2*pi)^{-1/2}/nb_obs/h)
}
second_term <- 2/(nb_obs-1)*sum(terms)
```