MANA130083.01 Nonparametric *Spring 2023*

Solution to Homework 6

Date: April 27, 2023 *Scribe: Jianliang He*

***Warning**: This note is only used as a reference solution for the homework, and the solution to each question is not unique. The solution may contain factual and/or typographic errors and comments and criticism are kindly welcomed.*

Problem 1 *Get the Sloan Digital Sky Survey data from* `https://www.stat.cmu.edu/~larry/all-of-nonpar/=data/galaxy.dat` *Investigate the distribution of redshifts using a histogram and a kernel density estimator. Use least squares cross-validation to choose the amount of smoothing. Also consider the Normal reference rule for picking a bandwidth for the kernel. Plot the graphs of the estimated densities. Carefully read example 4.3 from the textbook.*

Solution:

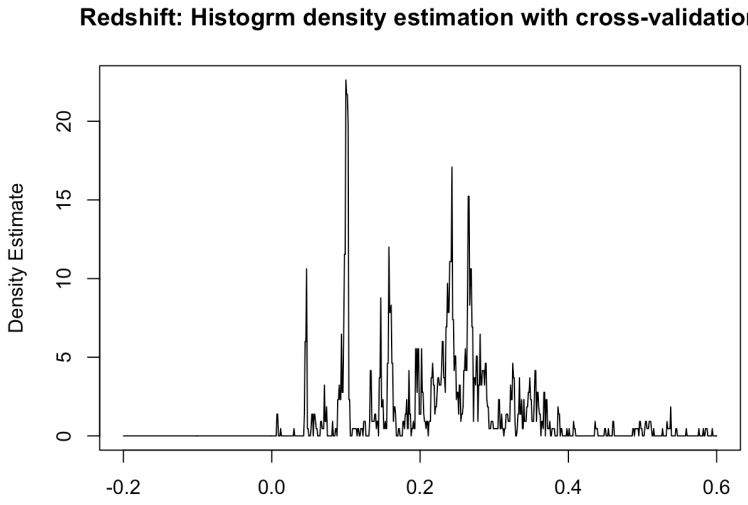
```
# Load data
data <- read.table
("https://www.stat.cmu.edu/~larry/all-of-nonpar/=data/galaxy.dat")[,3]

estden.hist <- function(x,data,h){
  n <- length(data)
  range <- max(data)-min(data)
  block <- floor((x-min(data))/h)
  p.hat <- sum(data>=min(data)+range*(block-1)*h &
            data<min(data)+range*block*h)/n
  return(p.hat/h)
}
```

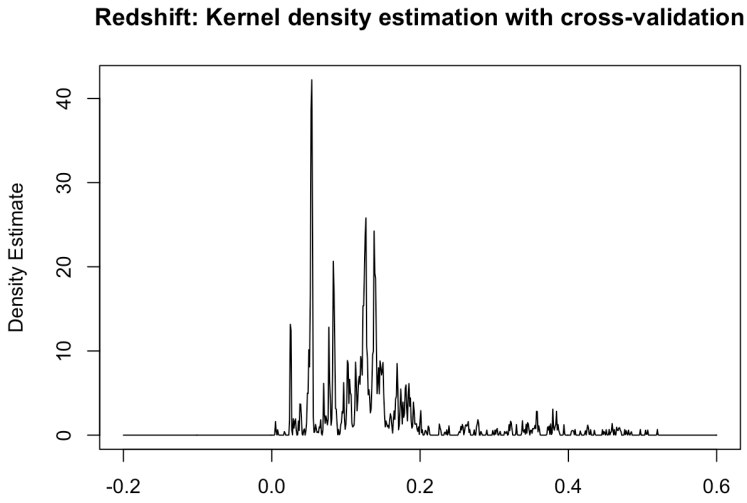
```r

band.hcv <- function(data){
  h.candidate <- seq(from = 0.00002, to = 0.01, by = 0.0001)
  n <- length(data)
  nh <- length(h.candidate)
  range <- max(data)-min(data)
  riskseq <- c()
  for (i in 1:nh){
    h <- h.candidate[i]
    p.hat = c()
    for (j in 1:n){
      p.hat[j] = sum(data>=min(data)+range*(j-1)*h
                     & data<min(data)+range*j*h)/n
    }
    # Empirical risk (6.16)
    riskseq[i] <- 2/h/(n-1)-(n+1)/h/(n-1)*sum(p.hat^2)
  }
  return(h.candidate[which.min(riskseq)])
}

estden.kernel <- function(x, data, h){
  n <- length(data)
  kernel.RBF <- function(x) exp(-(x^2)/2) / sqrt(2 * pi)
  return(sum(kernel.RBF((x - data)/h)) / (n*h))
}

band.kcv <- function(data){
  h.candidate <- seq(from = 0.00002, to = 0.01, by = 0.0001)
  n <- length(data)
  nh <- length(h.candidate)
```

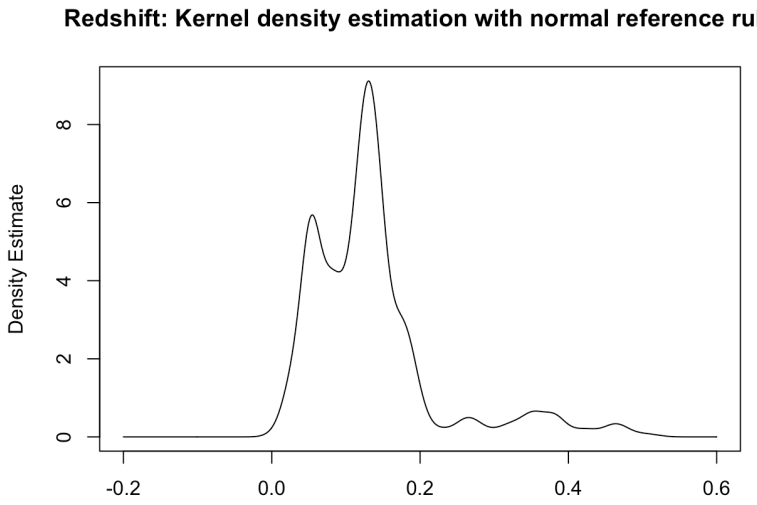
```
  riskseq <- c()
  for (i in 1:nh){
    h <- h.candidate[i]
    # Integrated square estimation
    estdensq.kernel <- function(x) estden.kernel(x, data = data, h = h)^2
    J1 <- integrate(Vectorize(estdensq.kernel),
                    lower = -1, upper = 2, subdivisions = 2000)$value
    # leave-one-out
    J2 <- 0
    for (j in 1:n){
      J2 <- J2 + estden.kernel(data[j], data = data[-j], h = h)
    }
    # Empirical risk (6.33)
    riskseq[i] <- J1 - (2 * J2)/n
  }
  return(h.candidate[which.min(riskseq)])
}

band.normal <-function(data){
  n <- length(data)
  s <- sd(data)
  qdata <- as.vector(quantile(data,probs=c(0.25,0.75)))
  Q <- qdata[2]-qdata[1]
  sigma <- min(s, Q/1.34)
  h <- 1.06*sigma*n^(-1/5)
  return(h)
}

# Density estimation
x.grid <- seq(from = -0.2, to = 0.6, by = 0.001)
```

```r
# Histogram : Cross-Validation
h.hcv <- band.hcv(data)
y.grid <- sapply(x.grid, estden.hist, data = data, h = h.hcv)
plot(x.grid, y.grid, type = "l", xlab = "", ylab = "Density Estimate",
     main = "Redshift: Histogrm density estimation with cross-validation")

# Kernel: Cross-Validation
h.kcv <- band.kcv(data)
y.grid <- sapply(x.grid, estden.kernel, data = data, h = h.kcv)
plot(x.grid, y.grid, type = "l", xlab = "", ylab = "Density Estimate",
     main = "Redshift: Kernel density estimation with cross-validation")

# Kernel: Normal reference rule
h.normal <- band.normal(data)
y.grid <- sapply(x.grid, estden.kernel, data = data, h = h.normal)
plot(x.grid, y.grid, type = "l", xlab = "", ylab = "Density Estimate",
     main = "Redshift: Kernel density estimation with normal reference rule")
```

Detailed plots for each density estimation can be found in next page. □


Redshift: Histogrm density estimation with cross-validation
Density Estimate
0
5
10
15
20
-0.2
0.0
0.2
0.4
0.6


Redshift: Kernel density estimation with cross-validation
Density Estimate
0
10
20
30
40
-0.2
0.0
0.2
0.4
0.6


Redshift: Kernel density estimation with normal reference rule
Density Estimate
0
2
4
6
8
-0.2
0.0
0.2
0.4
0.6