Exercise 1.

For non-parametric density estimation , by kernel method

我們有$\mathrm{MISE}=\mathrm{E(ISE)}=\mathrm{E}\left(\int\left(\hat{f}-f\right)^2\mathrm{dx}\right)=\int Var\left(\hat{f}\right)dx+\int Bias\left(\hat{f}\right)^2dx$

$$=\frac{b_n^4u_2^2}{4}R(f'')+\frac{1}{nb_n}R(k)+o\left(\frac{1}{nb_n}\right)$$

with optimal bandwith $\mathrm{b}_\mathrm{n}^*=\left(\frac{R(k)}{nu_2^2R(f'')}\right)^{\frac{1}{5}}$, where $\mathrm{u}_2=\int xk^2(x)dx$

$$,R(k)=\int k^2(x)dx\quad ,R(f'')=\int f''(x)^2dx\ .$$

這裡選擇 Eparechinlcov kernel,

$$\mathrm{k(x)}=\frac{3}{4}(1-x^2)\ ,\qquad -1<\mathrm{x}<1$$

經過計算(附錄), $\mathrm{u}_2^2=\frac{1}{25}$ $,R(f'')=0.2115626$ $,\mathrm{R(k)}=\frac{3}{5}$, 代入 MISE 中,

得到不同樣本數$\mathrm{n}_1=10$, $\mathrm{MISE}_1=0.2735451$

$\mathrm{n}_2=10^2$, $\mathrm{MISE}_2=0.04335398$

$\mathrm{n}_3=10^3$, $\mathrm{MISE}_3=0.006871143$

$\mathrm{n}_4=10^6$, $\mathrm{MISE}_4=0.00002735451$

$\mathrm{n}_5=10^7$, $\mathrm{MISE}_5=0.000004335398$

$\mathrm{n}_6=10^8$, $\mathrm{MISE}_6=0.0000006871143$

觀察到, $\frac{MISE_1}{MISE_4}=\frac{0.2735451}{0.00002735451}=10^4$ as $\mathrm{n}_4=10^5n_1$, 同樣$(\mathrm{n}_2,\mathrm{n}_5)$和$(\mathrm{n}_3,\mathrm{n}_6)$如上

故, as $\mathrm{n}\to\infty$ $,\mathrm{AMISE}=\mathrm{O}\left(\mathrm{n}^{-\frac{4}{5}}\right)$.

Exercise 2.

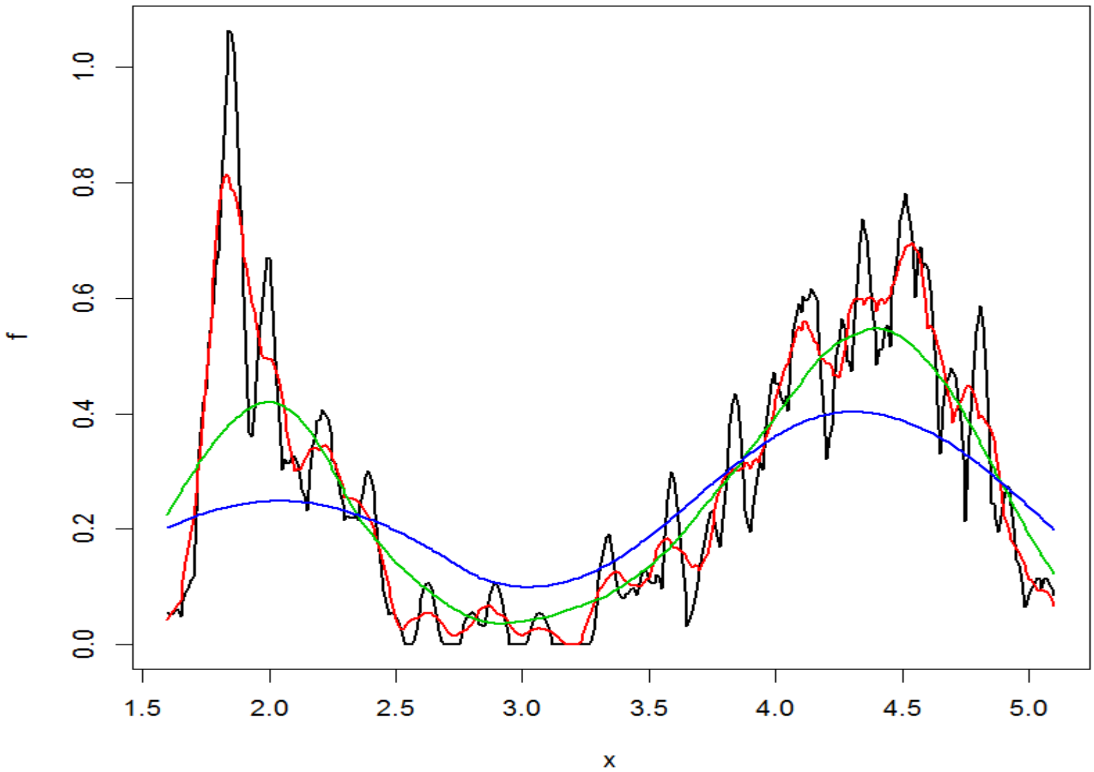
Data “faithful” 為黃石公園的噴泉噴發時間(X1)與等待時間(X2)，我們想要找到 X1 和 X2 的邊際密度函數$\mathrm{g}_{X_1}(x_1),\mathrm{g}_{X_2}(x_2)$和聯合密度函數$\mathrm{g}_{X_1X_2}(x_1,x_2)$.

利用 kernel method, 這裡選擇 Eparechinlcov kernel,

$$\mathrm{k(x)}=\frac{3}{4}(1-x^2)\ ,\qquad -1<\mathrm{x}<1$$

For $\mathrm{g}_{X_1}(x_1)$,

(i)我們有 $\mathrm{X}_1,X_2,\dots,X_{272}\ \sim\ \mathrm{g}_{X_1}$, 考慮估計量 $\mathrm{f}_n(x)=\frac{1}{nb_n}\sum_{i=1}^n k\left(\frac{x-X_i}{b_n}\right)$ 取 $\mathrm{n}=272$, take $\mathrm{b}_n=(0.05\ ,0.1\ ,0.5\ ,1)$, 對於不同的b_n估計畫圖如下:

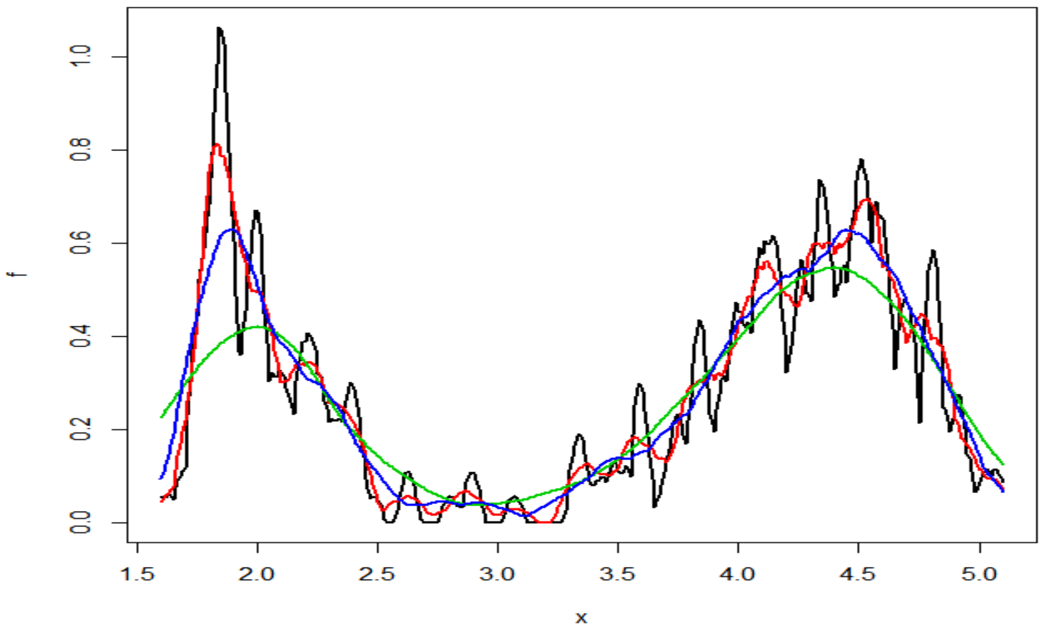


(黑色$\mathrm{b}_n = 0.05$;紅色$\mathrm{b}_n = 0.1$;綠色$\mathrm{b}_n = 0.5$;藍色$\mathrm{b}_n = 1$)

發現b_n愈小則圖形越曲折複雜，b_n愈大則圖形越平滑.

(ii)Choose b_n by peseudo likelihood,

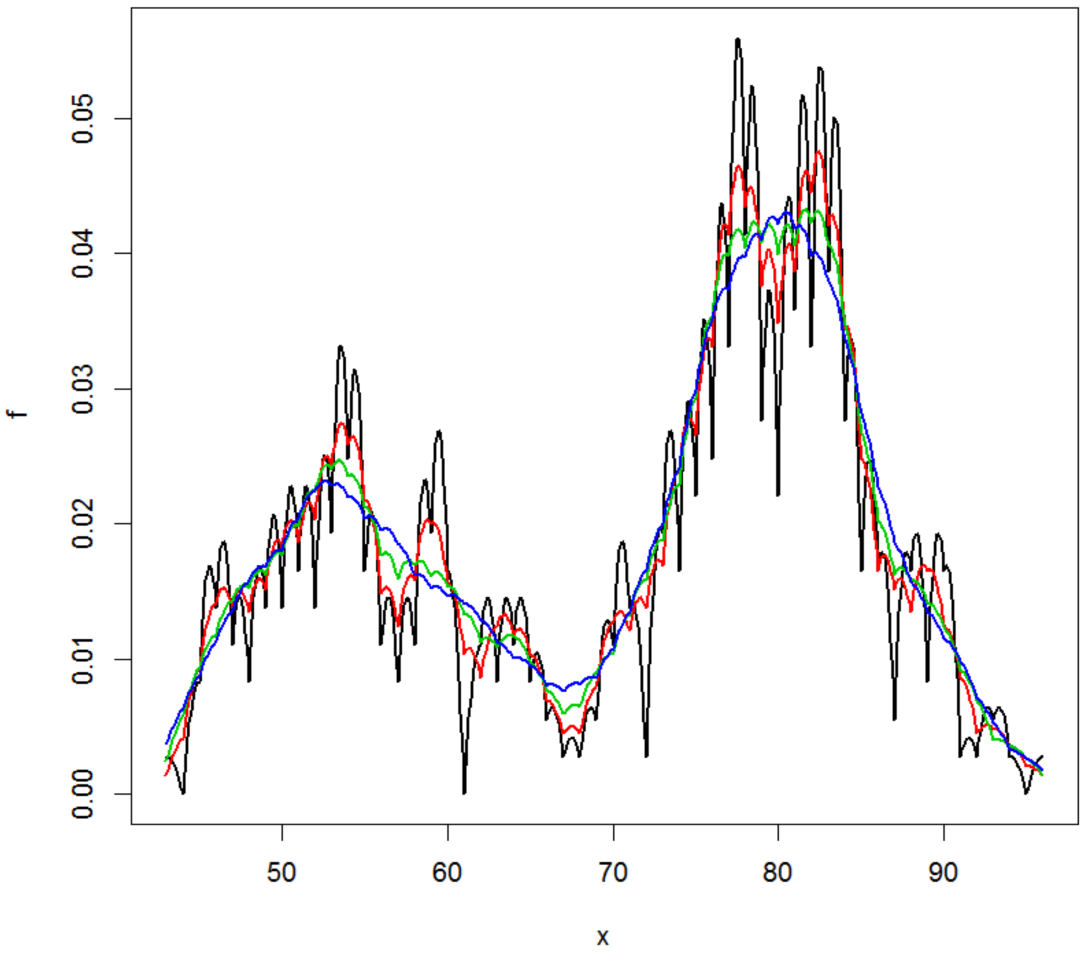
考慮$\mathrm{b}_\mathrm{n}^* = argmax_{b_n} p(b_n) = \prod_{i=1}^{n} \hat{f}_{-i}(x_i) = 0.21$，畫圖入下:



(黑色為$b_n = 0.05$，紅色為$b_n = 0.1$，綠色為$b_n = 0.5$，藍色為$b_n = 0.21$)

For $g_{X_2}(x_2)$,

我們有 $X_1, X_2, \ldots, X_{272} \sim g_{X_2}$，考慮估計量 $f_n(x) = \frac{1}{nb_n}\sum_{i=1}^{n} k\left(\frac{x-X_i}{b_n}\right)$ with $n = 272$，take $b_n = (1\ ,2\ ,3\ ,4)$，對於不同的b_n估計畫圖如下:



(黑色$b_n = 1$;紅色$b_n = 2$;綠色$b_n = 3$;藍色$b_n = 4$)

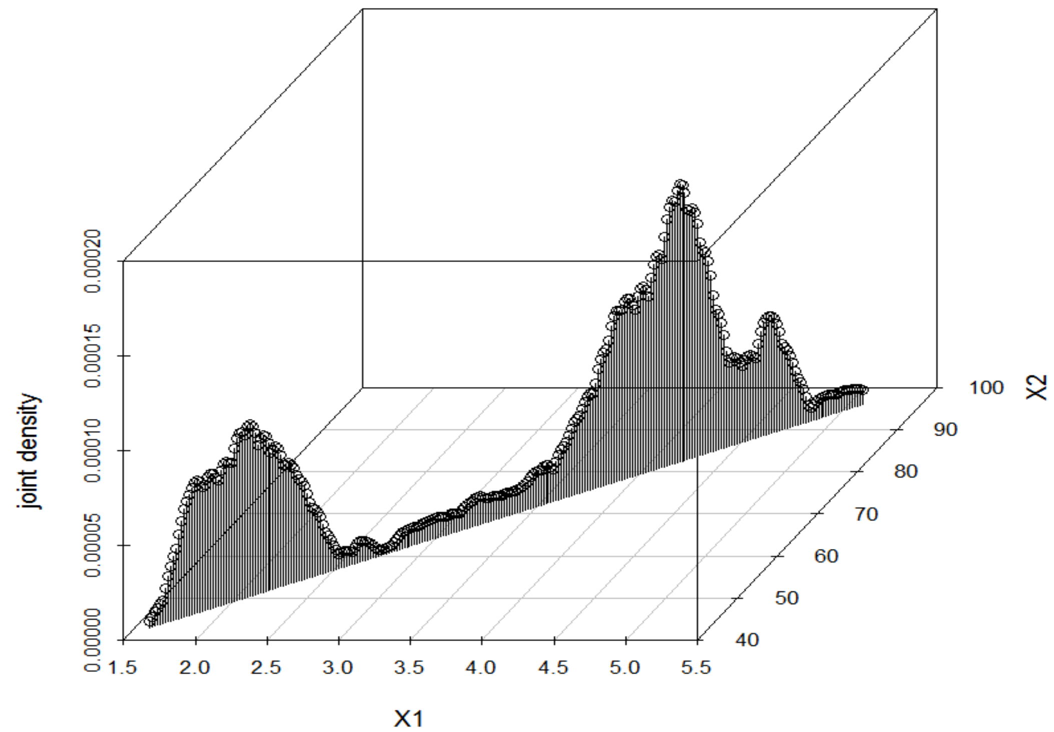
發現當$b_n = 4$圖形才比較平滑，應該是原始資料 X2 等待時間的尺度和 X1 噴發時間不同，可以看到不同樣本的等待時間差異較大. 故對於 bandwith 的選擇數字相對要大一些才趨於平滑.

For $g_{X_1X_2}(x_1, x_2)$,

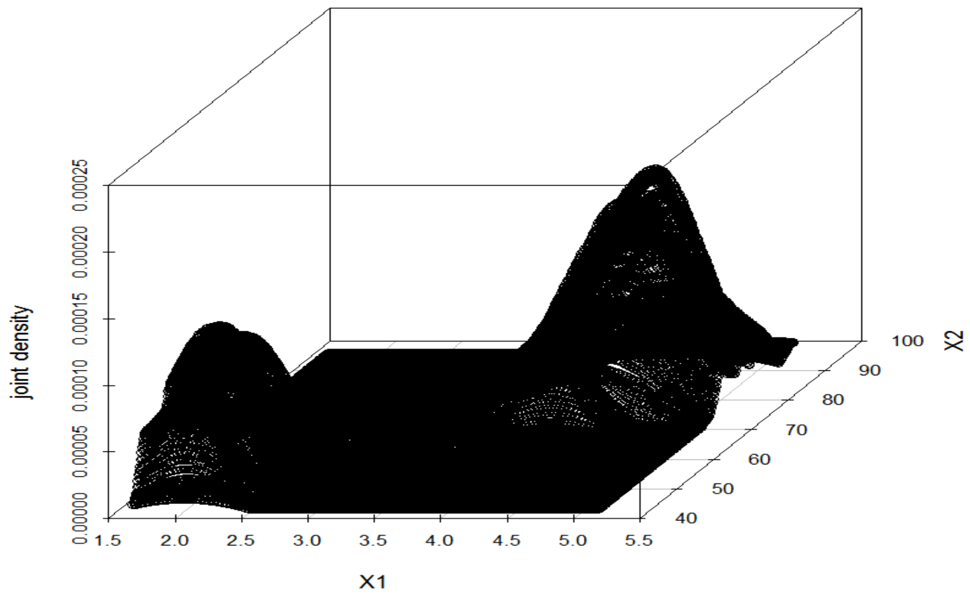
有$(X_{11}, X_{12}), (X_{21}, X_{22}), (X_{31}, X_{32}), \ldots, (X_{272,1}, X_{272,2}) \sim g_{X_1X_2}$，考慮估計量

$$f_n(x) = \frac{1}{n}\sum_{i=1}^{n}\prod_{j=1}^{p}\frac{1}{b_{nj}}k\left(\frac{x_j - X_{ij}}{b_{nj}}\right) \quad \text{, with } n = 272$$

, take $b_n = (b_{n1}, b_{n2}) = (0.5\ \ ,2)$, 畫出的 3 維圖如下:



(只畫出(X_1, X_2)對角線部分的$g_{X_1X_2}$)



(所有的(X_1, X_2)的$g_{X_1X_2}$)

#Exercise 1.

```
#non-parametric density est of kernal method
#R(f'')
R_f<-function(x){ 1/pi* (x^2-1)^2 *exp(-x^2) }
I_Simpson<-function(a,b,l){
   #a<-0;b<-20;l<-50
   h<-seq(a,b,(b-a)/l)
   if(l%%2==0){
      z1<-0;z2<-0
      for(i in 2:(l/2)){      z1<-z1+2*R_f(h[2*i-1])    }
      for(i in 1:(l/2)){      z2<-z2+4*R_f(h[2*i])    }
      I_S<-(R_f(h[1])+R_f(h[l+1])+z1+z2)*(b-a)/l/3
   } else {print("n need to be even")} }
Rf<-I_Simpson(0,20,50)
#R(k) with Eparechinlcov kernal
Rk<-3/5
#optimal bandwith b_n
bandwidth<-function(n){ ( Rk / n *(1/25) *Rf )^(1/5) }
#MISE
MISE<-function(n){ bandwidth(n)^4 *(1/25) *Rf /4 + 1/(n*bandwidth(n)) *Rk }
a<-MISE(10^6)
```

#Exercise 2.

```
#Eparechinlcov kernel
K<-function(x){      if(x>-1 && x<1){ return(3/4*(1-x^2)) }else{ return(0) } }
#Triangle kernel
K<-function(x){ if(x>-1 && x<1){ return(1-abs(x)) }else{ return(0) } }
#Uniform kernel
K<-function(x){ if(x>-1 && x<1){ return(1/2) }else{ return(0) } }
#Normal kernel
K<-function(x){ return(1/sqrt(2*pi)*exp(-x^2/2)) }
#g1 density
data<-faithful
n<-length(data[,1])
b_n<-c(0.05,0.1,0.5,1)
for(k in 1:length(b_n)){
   x<-seq(min(data[,1]),max(data[,1]),0.01)
   f<-rep(0,length(x))
   for(l in 1:length(x)){
      y<-rep(0,n)
      for(i in 1:n){
         y[i]<-K( (x[l]-data[i,1]) /b_n[k] ) /n /b_n[k]
         f[l]<-sum(y)
      }
   }
   if(k==1){ windows();plot(x,f,type = "l",lwd=2) }
   else{ lines(x,f,col= k,lwd=2) }
}
#g2 density
data<-faithful
n<-length(data[,2])
b_n<-c(1,2,3,4)
for(k in 1:length(b_n)){
   x<-seq(min(data[,2]),max(data[,2]),0.1)
   f<-rep(0,length(x))
   for(l in 1:length(x)){
      y<-rep(0,n)
      for(i in 1:n){
         y[i]<-K( (x[l]-data[i,2]) /b_n[k] ) /n /b_n[k]
         f[l]<-sum(y)
```

```
        }
    }
    if(k==1){ windows();plot(x,f,type = "l",lwd=2) }
    else{ lines(x,f,col= k,lwd=2) }
}
#g(x1,x2) density
data<-faithful
n<-272
#j=1
b_n1<-0.5
x1<-seq(min(data[,1]),max(data[,1]), (max(data[,1])-min(data[,1]))/n )
#j=2
x2<-seq(min(data[,2]),max(data[,2]), (max(data[,2])-min(data[,2]))/n )
b_n2<-2
#g(x1,x2) density  對角線部分
n<-272
x<-cbind(x1,x2)
f<-rep(0,length(x[,1]))
for(l in 1:length(x[,1])){
    y<-rep(0,n)
    for(i in 1:n){
        y[i]<-K( (x[l,1]-data[i,1]) /b_n1 ) /b_n1 /n *K( (x[l,2]-data[i,2]) /b_n2 ) /b_n2 /n
        f[l]<-sum(y)
    }
}
windows();plot(x[,1],f,type = "l",lwd=2)
windows();plot(x[,2],f,col = 2,type = "l",lwd=2)
library(scatterplot3d)
windows();scatterplot3d(x1,x2,f,color = ,type = "h",
                                            xlab = "X1",
                                            ylab = "X2",
                                            zlab = "joint density")
library(plotly)
plot_ly(data.frame(x=x1 ,y=x2 ,z=f)
                                    , x = ~x ,y = ~y, z = ~z,color = , colors = ) %>%
    add_markers() %>%
    layout(scene = list(xaxis = list(title = "X1"),
                                        yaxis = list(title = "X2"),
```

```
                                           zaxis = list(title = "joint density")))
#3 維圖全部
n<-272
f<-matrix(0,length(x2),length(x1))
for(k in 1:length(x1)){
  for(l in 1:length(x2)){
    y<-rep(0,n)
    for(i in 1:n){
      y[i]<-K( (x1[k]-data[i,1]) /b_n1 ) /b_n1 /n *K( (x2[l]-data[i,2]) /b_n2 ) /b_n2
/n
      f[k,l]<-sum(y)
    }
  }
}
plot(x=1:9,y=matrix(11:19,3,3))
B<-c()
for(i in 1:273){
  A<-cbind(x1,x2[i])
  B<-rbind(B,A)
}
windows();scatterplot3d(B[,1],B[,2],f,color = ,type = "p",
                                            xlab = "X1",
                                            ylab = "X2",
                                            zlab = "joint density")
plot_ly(data.frame(x=B[,1] ,y=B[,2] ,z=as.vector(f) )
        , x = ~x ,y = ~y, z = ~z,color = , colors = ) %>%
  add_markers() %>%
  layout(scene = list(xaxis = list(title = "X1"),
                                        yaxis = list(title = "X2"),
                                        zaxis = list(title = "joint density")))
```