EXERCISE 7

WEIYU LI

1. For the *Bart Simpson* density in Page 6 in the slides, that is,

$$f(x) = \frac{1}{2}\phi(x;0,1) + \frac{1}{10}\sum_{j=0}^{4}\phi(x;j/2-1,1/10),$$

where $\phi(x;\mu,\sigma)$ is the normal pdf with mean μ and standard deviation σ. Generate 1000 random numbers, estimate its density by histogram and compare different bins. Use naive density estimator to estimate density, draw the figures, and compare the influences of different bandwidth h.

Hint. Note that $Y \sim f$ has the same distribution of UX where $U \sim U(0,1)$ and $X|_{U>0.5} \sim N(0,1), X|_{0.1j<U\leq 0.1(j+1)} \sim N(j/2-1,1/10^2)$. The following codes are an example.

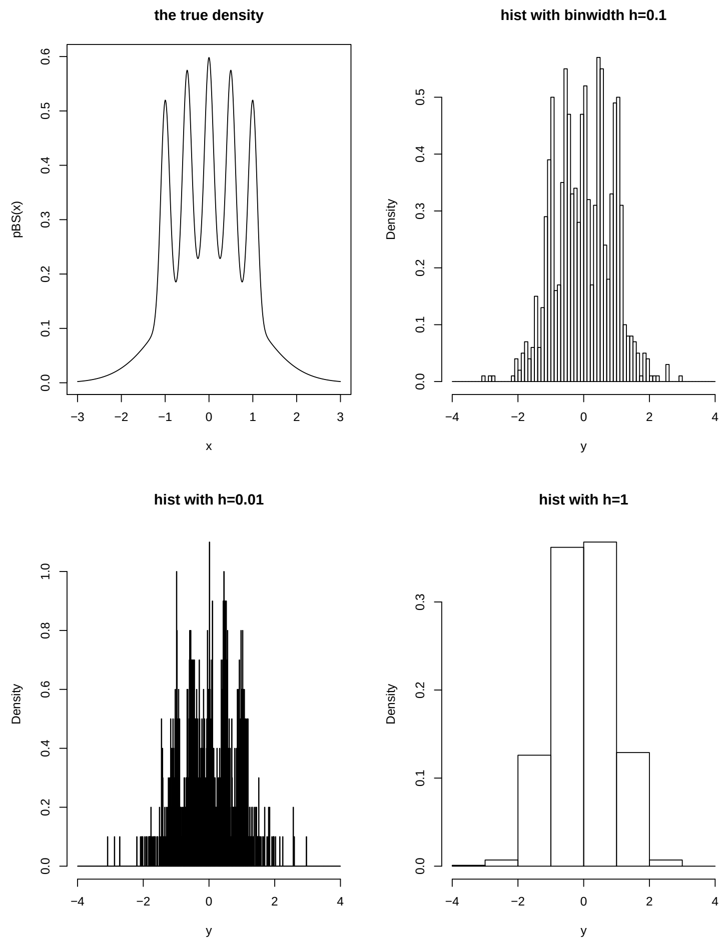
```
set.seed(0)
# true density
pBS <- function(x){ # probability of BS density
  f <- 1/2 * dnorm(x, 0, 1)
  for (j in 0:4) {
    f <- f + 1/10 * dnorm(x, j/2-1, 1/10)
  }
  return(f)
}
x <- seq(-3, 3, 0.01)
par(mfrow=c(2,2))
plot(x, pBS(x), 'l', main = 'the true density')

# randomly generating
rBS <- function(n){ # random points from BS density
  u <- runif(n)
  y <- u
  ind <- which(u > 0.5) #index for those generated from N(0,1)
  y[ind] <- rnorm(length(ind), 0, 1)
  for (j in 0:4) {
    ind <- which(u > j * 0.1 & u <= (j+1) * 0.1)
    #index for those generated from N(j/2-1,1/10^2)
    y[ind] <- rnorm(length(ind), j/2 -1, 1/10)
  }
  return(y)
}
n <- 1000
y <- rBS(n)
```

Date: 2019/10/28.

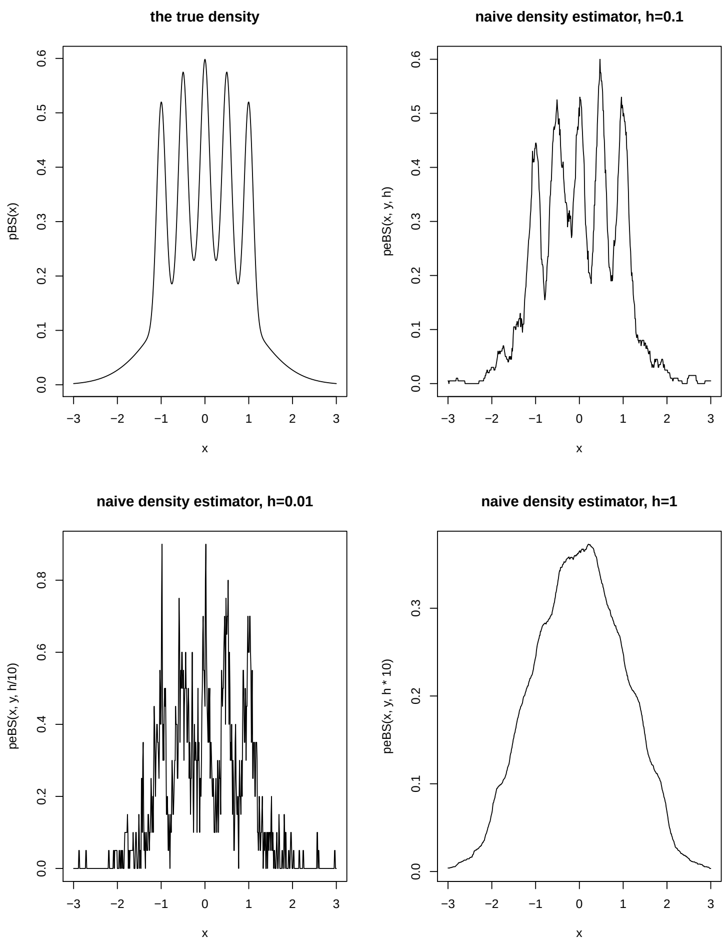
liweiyu@mail.ustc.edu.cn.

```
h <- 0.1
hist(y, breaks = seq(-4, 4, h), probability = TRUE,
main = 'hist with binwidth h=0.1')
hist(y, breaks = seq(-4, 4, h / 10), probability = TRUE, main = 'hist with h=0.01')
hist(y, breaks = seq(-4, 4, h * 10), probability = TRUE, main = 'hist with h=1')
```




```
# naive estimator
peBS <- function(x, y, h){ # probability estimator of BS density
  # input: x - estimate points, y - samples, h - bandwidth
  # output: y - estimated density at x
  m <- length(x)
  n <- length(y)
  ye <- rep(0, m)
  for (i in 1:n){
    ye <- ye + as.numeric((x >= y[i] - h) & (x < y[i] + h))
  }
  ye <- ye / (2*h*n)
  return(ye)
}
plot(x, pBS(x), 'l', main = 'the true density')
```

```
plot(x, peBS(x, y, h), 'l', main = 'naive density estimator, h=0.1')
plot(x, peBS(x, y, h / 10), 'l', main = 'naive density estimator, h=0.01')
plot(x, peBS(x, y, h * 10), 'l', main = 'naive density estimator, h=1')
```



$\square$

2. Prove the asymptotic normal distribution (first point in Page 46), that is,

$$n^{2/5}\left(\hat{f}_h(x) - f(x)\right) \rightsquigarrow N\left(\frac{c^2}{2}f''(x)\kappa_{21}, \frac{1}{c}f(x)\kappa_{02}\right),$$

where $\kappa_{ij} = \int u^i K^j(u)du, h = cn^{-1/5}$**.**

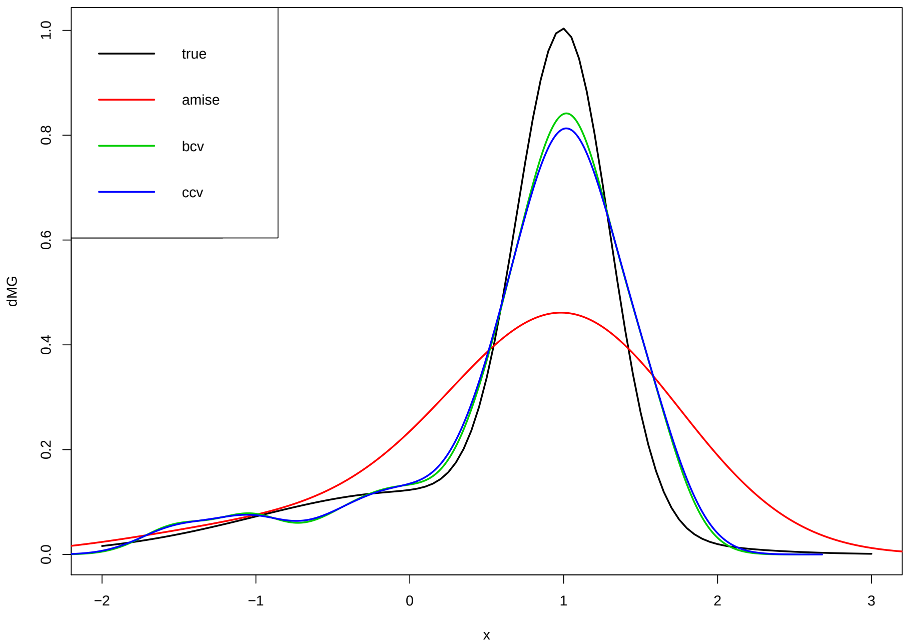
Proof. From the Theorem in Page 28, we know that if $h \to 0, nh \to 0\infty$, *then*

$$E(\hat{f}_h(x) - f(x)) = \frac{1}{2}h^2 f''(x)\kappa_{21} + o(h^2),$$
$$Var(\hat{f}_h(x)) = \frac{1}{nh}f(x)\kappa_{02} + o(\frac{1}{nh}).$$

Plugging in the value of h, and using central limit theorem completes the proof. $\square$

3. Generate 100 **data points from the distribution** $0.3N(0,1) + 0.7N(1,0.3^2)$**, use the bandwidth selection methods in R package** ***kedd*****, and draw the estimated density curves with different bandwidths in one plot.**

```
install.packages('kedd')
library(kedd)
set.seed(0)
n <- 100
dMG  <- function(x) 0.3 * dnorm(x,0,1) + 0.7 * dnorm(x,1,0.3)
x <- seq(-3, 3, 0.01)
rMG <- function(n){
  # randomly generate n points from the Mixed Gaussian distribution
  r <- runif(n, 0, 1)
  x <- r
  ind <- which(r < 0.3) #index for those generated from N(0,1)
  x[ind] <- rnorm(length(ind), 0, 1)
  x[-ind] <- rnorm(n-length(ind), 1, 0.3)
  return(x)
}
x <- rMG(n)
fhat.amise <- dkde(x, h = h.amise(x)$h) # BW selection: amise
fhat.bcv <- dkde(x, h = h.bcv(x)$h) # BW selection: bcv
fhat.ccv <- dkde(x, h = h.ccv(x)$h) # BW selection: ccv
# one can also use h.mcv/mlcv/tcv/ucv to get different bandwidth (BW)
plot(dMG, from = -2, to = 3, lwd = 2)
lines(fhat.amise$eval.points, fhat.amise$est.fx, col = 2, lwd = 2)
lines(fhat.bcv$eval.points, fhat.bcv$est.fx, col = 3, lwd = 2)
lines(fhat.ccv$eval.points, fhat.ccv$est.fx, col = 4, lwd = 2)
legend('topleft', legend = c('true', 'amise', 'bcv', 'ccv'),
       col = 1:4, lwd = c(2, 2, 2, 2))
```



□

4. Prove the second point in Page 44 in the slides: suppose now that we have estimated an unknown density f using some kernel K^A and bandwidth h_A, then we should use bandwidth

$$h_B = h_A \frac{\delta_0^B}{\delta_0^A}, \text{ where } \delta_0 = \left(\frac{\kappa_{02}}{\kappa_{21}^2}\right)^{1/5}$$

in the estimation with kernel K^B when we want to get approximately the same degree of smoothness as we had in the case of K^A and h_A.

Proof. Consider $K^A = K$, $K^B = K^c = \frac{1}{c}K(\frac{\cdot}{c})$. On the one hand, we have that using

$$h^c = \frac{1}{c}h$$

gives the same KDE. On the other hand, by change of variable formula, we obtain that

$$\kappa_{02}^c = \int \left(\frac{1}{c}K(\frac{u}{c})\right)^2 du = \frac{1}{c}\int (K(v))^2 dv = \frac{1}{c}\kappa_{02},$$
$$\kappa_{21}^c = \int u^2 \frac{1}{c}K(\frac{u}{c})du = c^2 \int v^2 K(v)dv = c^2\kappa_{02},$$

which gives that

$$\delta_0^c = \frac{1}{c}\delta_0.$$

In summary, $h^c = \frac{\delta_0^c}{\delta_0}h$ and more generally

$$h_B = h_A \frac{\delta_0^B}{\delta_0^A}$$

result in the same degree of smoothness. □

Another proof. We know from Page 41 that $AMISE(K_\delta) = \frac{\|K_\delta\|^2}{nh} + \frac{h^4}{4}\|f''\|^2 \mu_2^2(K_\delta)$ (or consider $AMSE$ from Page 29), which is minimized when

$$h^{K_\delta} = \left(\frac{\|f''\|^2}{n}\right)^{1/5} \delta_0^{K_\delta}.$$

Therefore, the optimal bandwidth is proportional to its corresponding δ_0. □