UK Complex Statistical Methods, WS 2024/25
Exercise sheet 1

Khodosevich Leonid

November 7, 2024

This was my first LaTeX experience, I hope everything will be rather understandable.

1 Problem 1

Kernel estimator of the s-th derivative $f^{(s)}$ of a density $f \in \mathcal{F}(\beta, L)$, $s < \beta$, with $f(x) \leq f_{\max} < \infty$ for all $x \in \mathbb{R}$, can be defined as follows

$$\widehat{f}^{(s)}(x; h) = \frac{1}{nh^{s+1}} \sum_{i=1}^{n} K\left(\frac{X_i - x}{h}\right),$$

where $h > 0$ is a bandwidth and $K : \mathbb{R} \to \mathbb{R}$ is a bounded kernel with support $[-1, 1]$ satisfying for $\ell = \lfloor \beta \rfloor$

$$\int u^j K(u)\, du = 0, \quad j = 0, 1, \ldots, s-1, s+1, \ldots, \ell,$$

$$\int u^s K(u)\, du = s!.$$

Prove that

(a) the bias of $\widehat{f}^{(s)}(x; h)$ is bounded by $c_1 h^{\beta - s}$ uniformly over the class $\mathcal{F}(\beta, L)$ for appropriate constant c_1 and a given point $x \in \mathbb{R}$; (2 points)

(b) the variance of $\widehat{f}^{(s)}(x; h)$ is bounded by $c_2 (nh^{2s+1})^{-1}$ uniformly over the class $\mathcal{F}(\beta, L)$ for appropriate constant c_2 and a given point $x \in \mathbb{R}$; (2 points)

(c) the maximum of the mean squared error of $\widehat{f}^{(s)}(x; h)$ over $\mathcal{F}(\beta, L)$ is of order $O(n^{-2(\beta - s)/(2\beta + 1)})$ if the bandwidth $h = h_n$ is chosen optimally. (2 points)

Solutions

I will start with b) as this is the way we've done this in lectures

Proof

b) Let

$$Z_i(x) = K\left(\frac{X_i - x}{h}\right) - \mathbb{E}\left\{K\left(\frac{X_i - x}{h}\right)\right\}.$$

$Z_1(x), \ldots, Z_n(x)$ are i.i.d. random variables with zero mean and variance

$$\begin{aligned}
\operatorname{var}\left\{\hat{f}^{(s)}(x;h)\right\} &= \mathbb{E}\left[\frac{1}{nh^{(s+1)}}\sum_{i=1}^{n} K(\frac{X_i - x}{h})\right] \\
&= \mathbb{E}\left[\frac{1}{nh^{(s+1)}}\sum_{i=1}^{n} K(\frac{X_i - x}{h}) - \frac{1}{nh^{(s+1)}}\sum_{i=1}^{n} \mathbb{E}K(\frac{X_i - x}{h})\right]^2 \\
&= \frac{1}{nh^{2s+2}}\mathbb{E}\left[\sum_{i=1}^{n} Z_i(x)\right]^2 \\
&= \frac{1}{nh^{2s+2}}\mathbb{E}Z_1^2(x) \, (\text{using } i.i.d. \text{ of } Z_i)
\end{aligned} \tag{1}$$

$$\begin{aligned}
\mathbb{E}\{Z_i(x)\}^2 &= \mathbb{E}\left[\left\{K\left(\frac{X_i - x}{h}\right)\right\}^2\right] - \left[\mathbb{E}\left\{K\left(\frac{X_i - x}{h}\right)\right\}\right]^2 \\
&\leq \mathbb{E}\left[\left\{K\left(\frac{X_i - x}{h}\right)\right\}^2\right] \, (\text{using that second term is } > 0) \\
&= \int \left\{K\left(\frac{u - x}{h}\right)\right\}^2 f(u)\, du \\
&\leq f_{\max} h \int \{K(x)\}^2 \, dx. \, (\text{using } \frac{u-x}{h} = z \text{ and } f(x) \leq f_{\max} < \infty)
\end{aligned} \tag{2}$$

Putting (2) in (1):

$$\begin{aligned}
\operatorname{var}\left\{\hat{f}^{(s)}(x;h)\right\} &\leq \frac{h}{nh^{2s+2}} f_{\max} \int \{K(x)\}^2 \, dx \\
&= \frac{1}{nh^{2s+1}} f_{\max} \int \{K(x)\}^2 \, dx \\
&= \frac{1}{nh^{2s+1}} c_2,
\end{aligned}$$

where

$$c_2 = f_{\max} \int \{K(x)\}^2 \, dx$$

$\blacksquare$

a)

$$\begin{aligned}
\text{bias}\left[\hat{f}^{(s)}(x;h)\right] &= \mathbb{E}\left[\hat{f}^{(s)}(x;h)\right] - f^{(s)}(x) = \\
&= \mathbb{E}\left[\frac{1}{nh^{s+1}} \sum_{i=1}^{n} K\left(\frac{X_i - x}{h}\right)\right] - f^{(s)}(x) = \\
&= \frac{1}{h^{s+1}} \int K\left(\frac{u-x}{h}\right) f(u)\, du - f^{(s)}(x) \,(\text{using linearity of}\, \mathbb{E}) = \\
&= \frac{1}{h^s} \int K\left(z\right) f(zh + x)\, dz - f^{(s)}(x) \,(z = \frac{u-x}{h}, du = hdz) = \\
&= \frac{1}{h^s} \int K\left(z\right) \left[f(x) + ... + \frac{(zh)^s}{s!} f^{(s)}(x) + ... + \frac{(zh)^l}{l!} f^{(l)}(x + zh\tau)\right] \, dz \\
-f^{(s)}(x)\,(Taylor) &= \\
&= \frac{1}{h^s} \int K(z) \frac{(zh)^s}{s!} f^{(s)}(x)\, dz + \frac{1}{h^s} \int K(z) \frac{(zh)^l}{l!} f^{(l)}(x + zh\tau)\, dz - f^{(s)}(x) \\
(\text{using conditions on K}) &= \\
&= \frac{1}{h^s} \frac{s!}{s!} h^s f^{(s)}(x) - f^{(s)}(x) + \frac{1}{h^s} \int K(z) \frac{(zh)^l}{l!} f^{(l)}(x + zh\tau)\, dz \\
(\text{using conditions on K}) &= \\
&= \frac{1}{h^s} \int K(z) \frac{(zh)^l}{l!} f^{(l)}(x + zh\tau)\, dz
\end{aligned}$$

Using that:

$$\begin{aligned}
\left|\text{bias}\left[\hat{f}^{(s)}(x;h)\right]\right| &\leq \frac{1}{l!h^s} \int \left|K(z)(zh)^l f^{(l)}(x + zh\tau) - f^{(l)}(x)\right| \, dz \\
&\leq \frac{1}{l!h^s} \int |K(z)|\, |zh|^l L |\tau zh|^{(\beta - l)}\, dz \,(\text{Hölder}) \\
&\leq \frac{L}{l!} h^{(\beta - s)} \int |K(z)||z|^\beta \, dz \\
&= c_1 h^{(\beta - s)}
\end{aligned}$$

where

$$c_1 = \frac{L}{l!} \int |K(z)||z|^\beta \, dz$$

$\blacksquare$

c)

$$\text{MSE}(\widehat{f}^{(s)}(x;h)) = c_1^2 h^{2(\beta-s)} + \frac{c_2}{nh^{2s+1}}$$

$$\begin{aligned}
\frac{d}{dh}\text{MSE}(\widehat{f}^{(s)}(x;h)) &= 2(\beta-s)c_1^2 h^{2(\beta-s)-1} - \frac{c_2 n^{-1}(2s+1)}{h^{2(s+1)}} \\
&= \frac{2(\beta-s)c_1^2 h^{2\beta+1} - c_2 n^{-1}(2s+1)}{h^{2(s+1)}}
\end{aligned}$$

Which yields:

$$h_{\text{MSE}} = \left[\frac{c_2(2s+1)}{2c_1^2(\beta-s)}\right]^{\frac{1}{2\beta+1}} n^{\frac{-1}{2\beta+1}}$$

And finally:

$$\begin{aligned}
&\text{MSE}(\widehat{f}^{(s)}(x;h_{\text{MSE}})) \\
&\qquad = c_2 n^{-1} C n^{\frac{2s+1}{2\beta+1}} + c_1 C n^{\frac{2\beta-2s}{2\beta+1}} \\
&\qquad = \widetilde{C} n^{-\frac{2(\beta-s)}{2\beta+1}} \\
&\qquad = O(n^{-\frac{2(\beta-s)}{2\beta+1}})
\end{aligned}$$

$\blacksquare$

Excercise sheet 1 (problems 2, 3)

Khodosevich Leonid

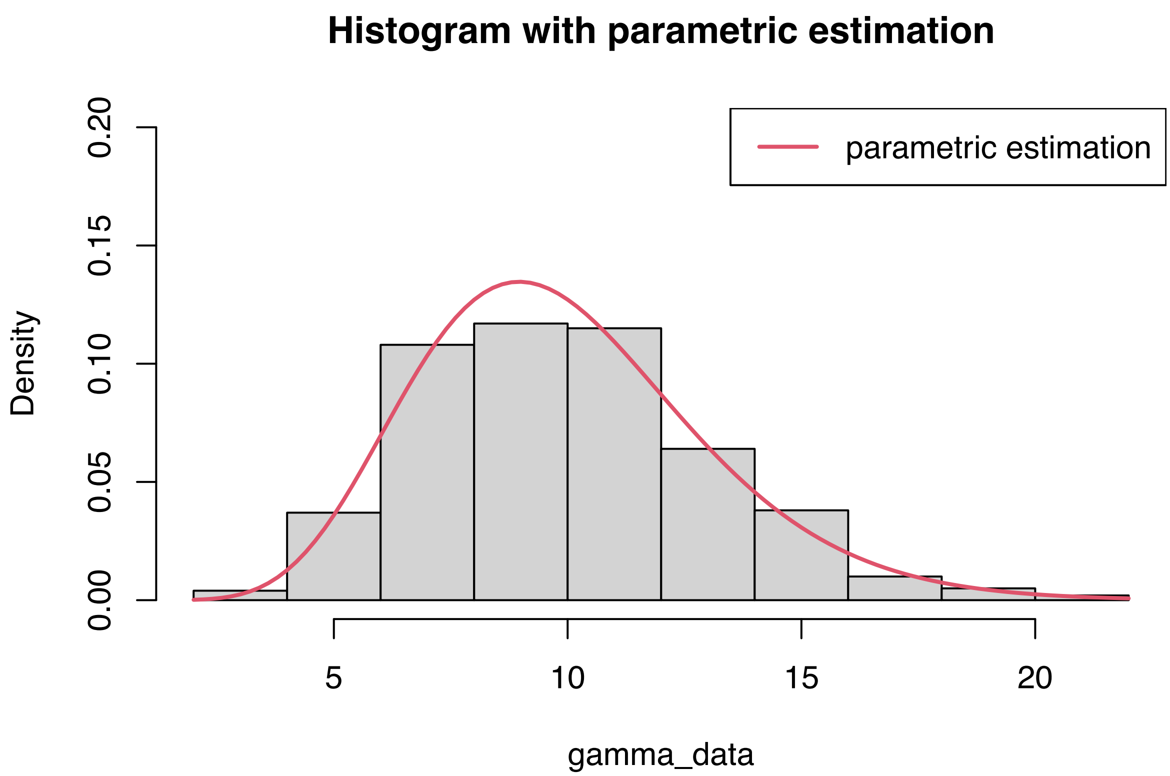
Problem 2

Subtask 1.

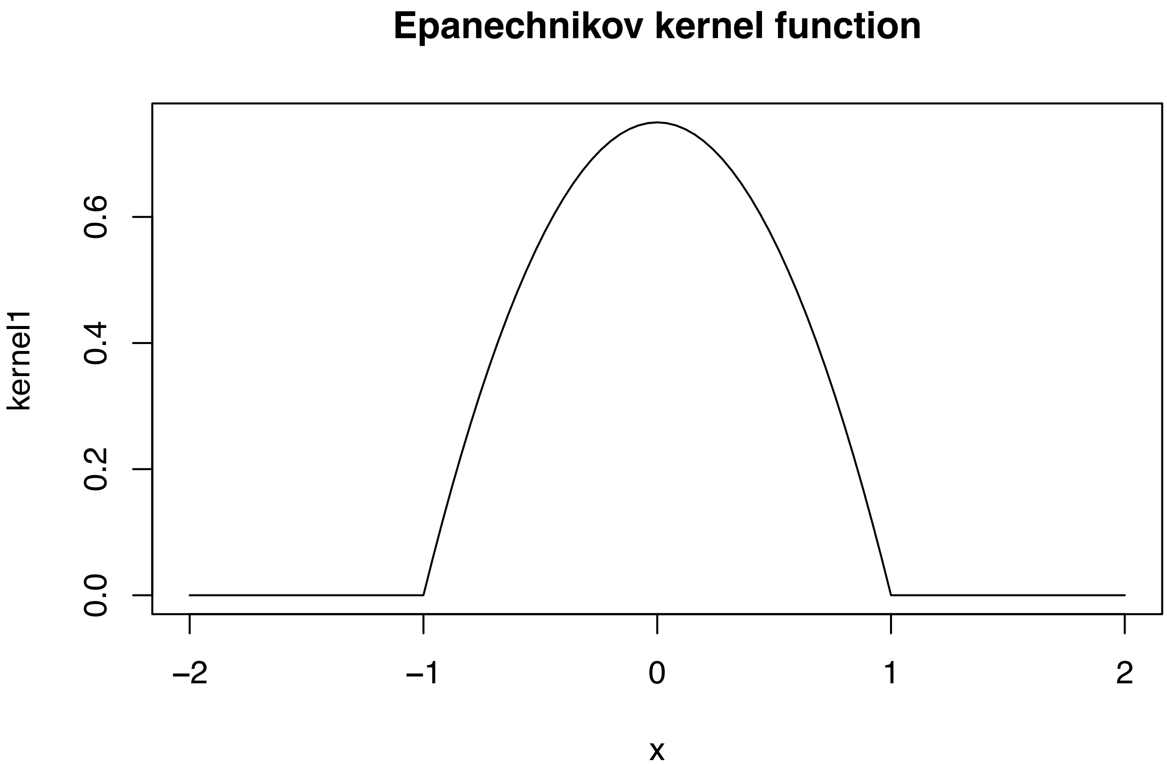
Simulate a dataset of 500 observations from the Gamma distribution with the shape pa- rameter 10 and the rate parameter 1. Set seed to 2425 to ensure comparability.

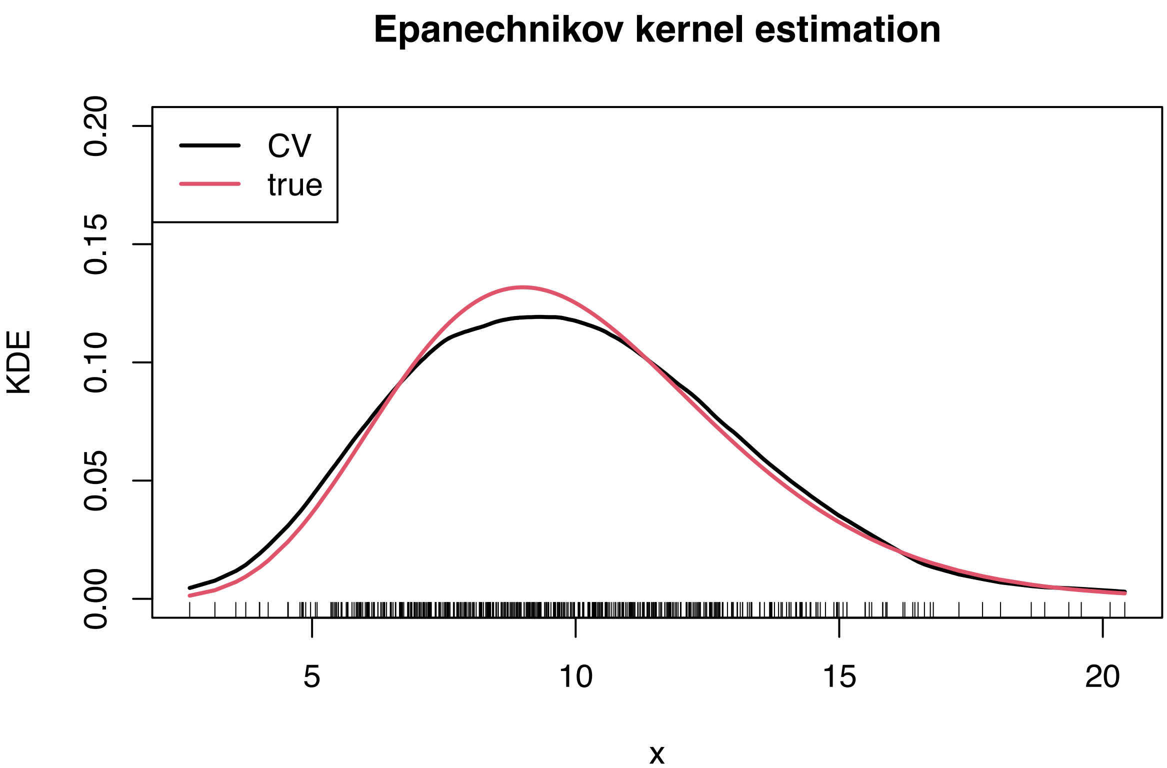
(a) Obtain a parametric estimator for the density of these data, employing moment estimators for the parameters. (1 point)
Estimators

$$alpha := \frac{Mean(X)^2}{Var(X)}, beta := \frac{Var(X)}{Mean(X)}$$

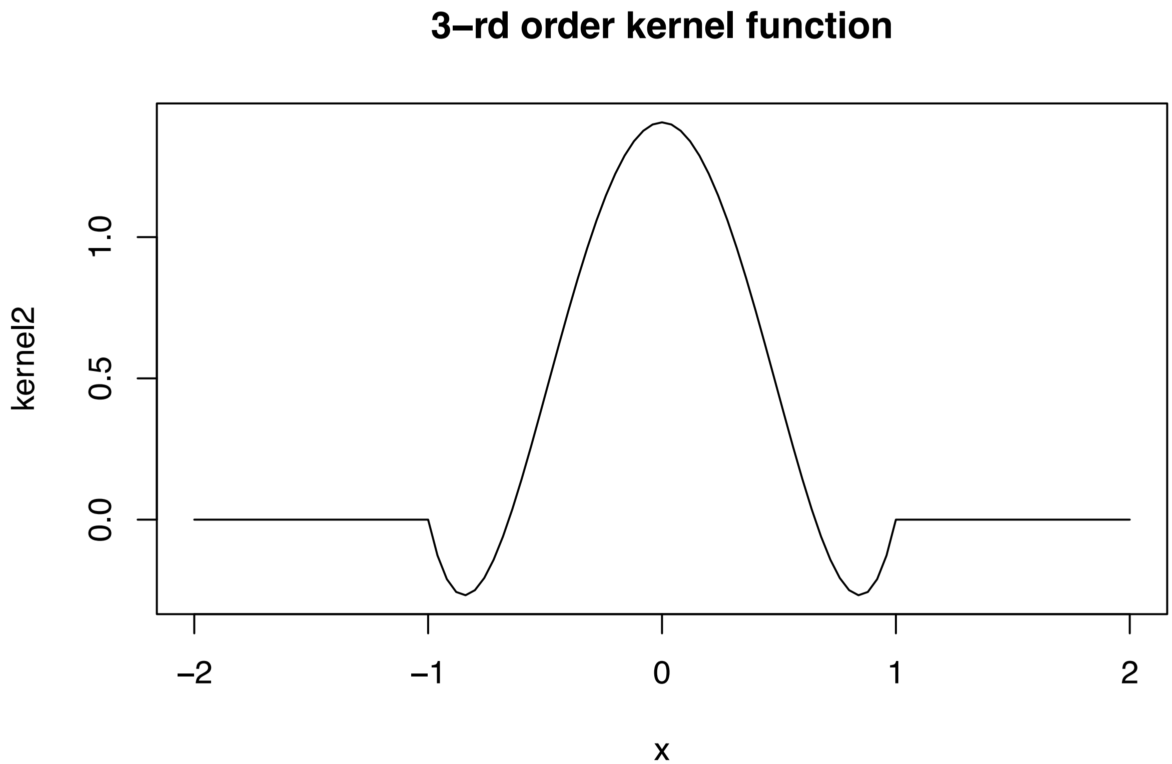


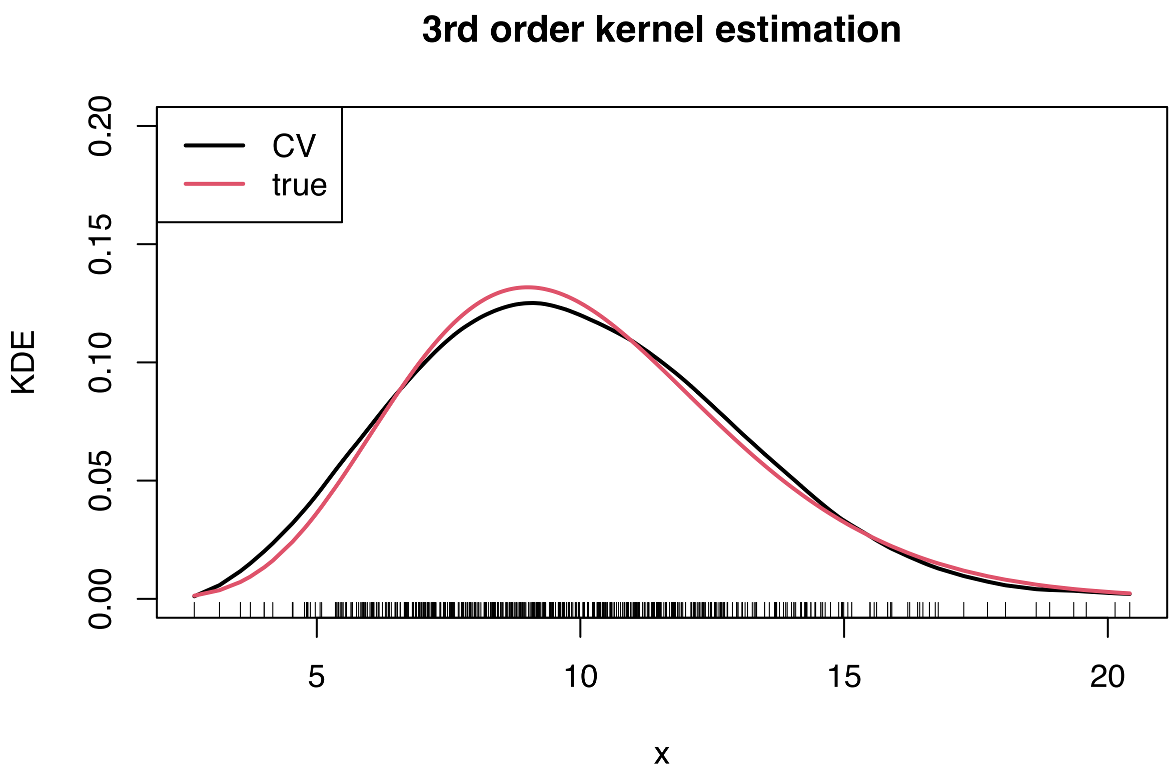
(b) Obtain a kernel density estimator with Epanechnikov kernel and a bandwidth chosen by cross-validation. Ensure that the cross-validation criterion has a global minimum by plotting it on a suitable range of bandwidths. (2 points)



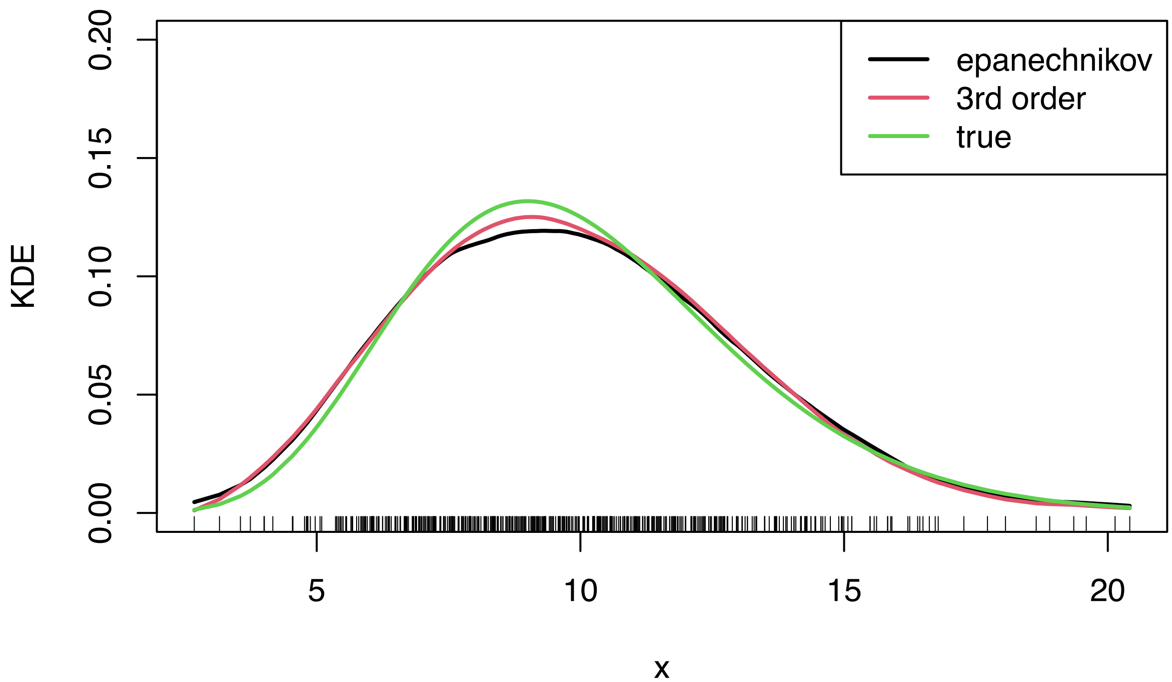


(c) Obtain a kernel density estimator as in (b), replacing Epanechnikov kernel by a kernel of order 3 (1 point)





(d) Plot estimators obtained in (a) – (c) together with the true density on one plot, putting a legend. Comment on the results. Compare also the bandwidths obtained in (b) and (c): which one is larger and why (give theoretical justification)? (2 points)




```
h1.cv
```

```
## [1] 2.307834
```

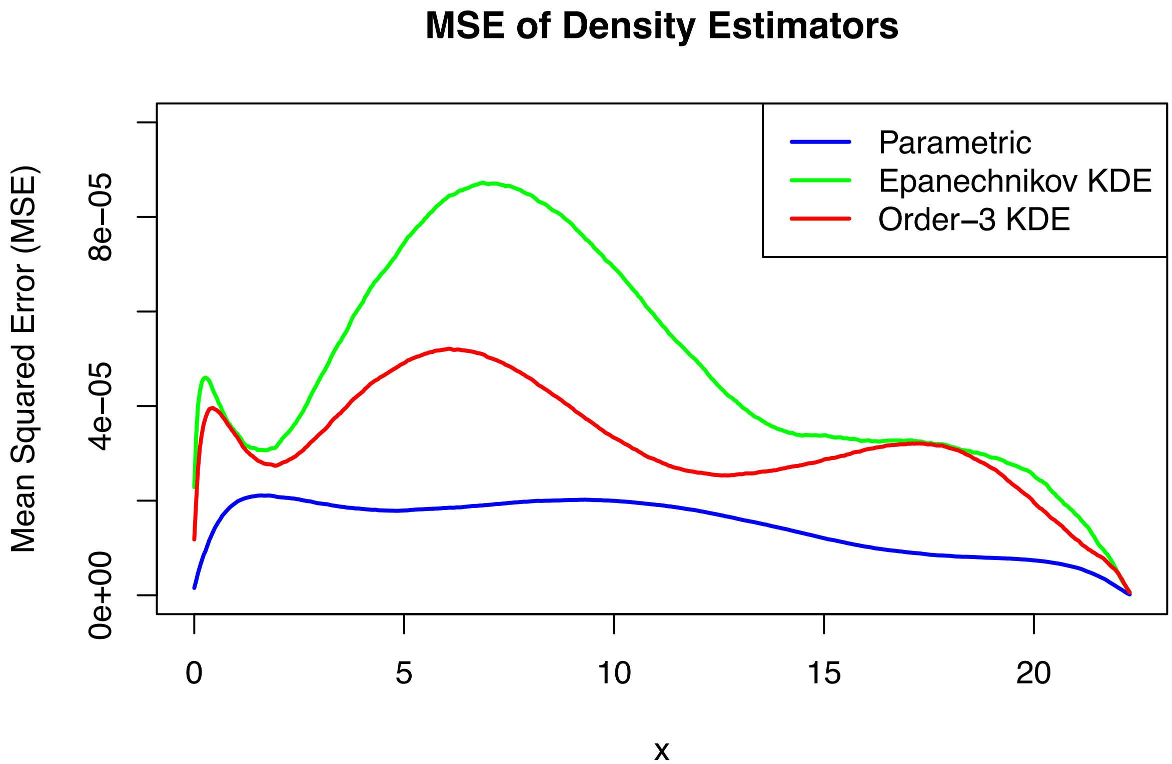
```
h2.cv
```

```
## [1] 4.931752
```

- 3rd order kernel has bigger smoothness, compared to 1st order epanechnikov kernel. For smoother kerels we need to have bigger bandwidth to prevent them from overfitting to data. If we would decrease brandwidth it would lead to decrease of bias term and increase of varience. kde would oscillate around true pdf.

Subtask 2

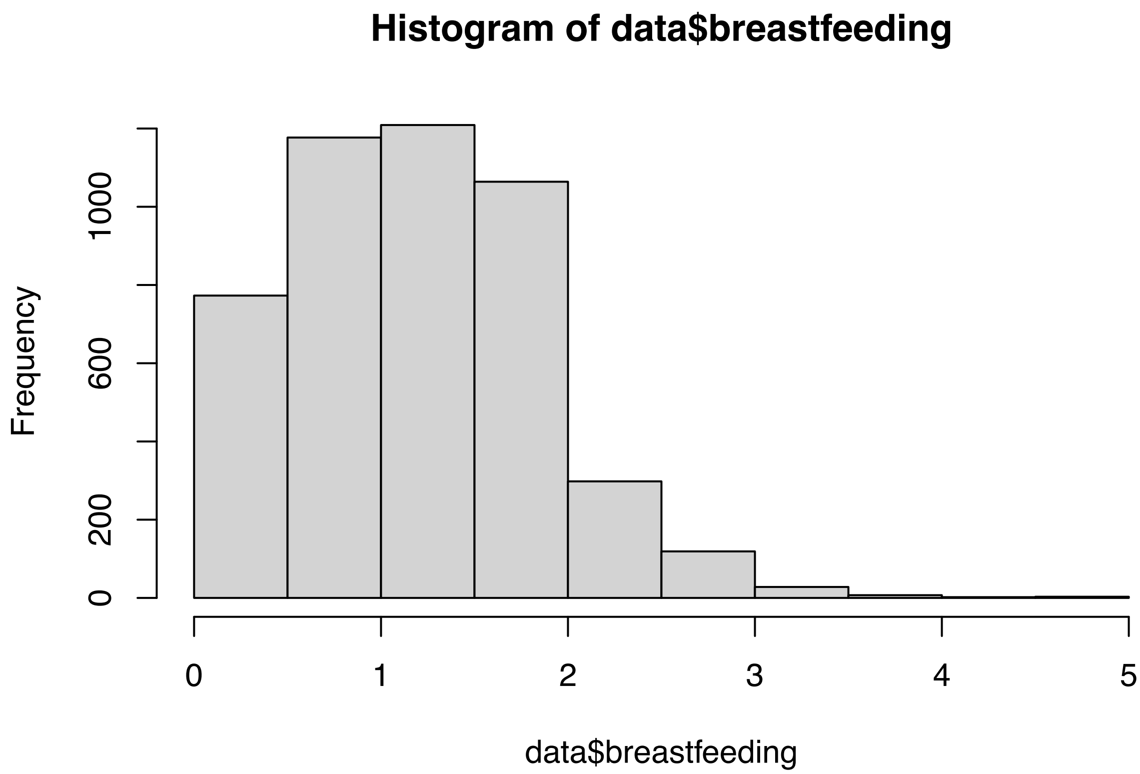
2. Simulate 300 samples as in 1, setting seed again to 2425. Estimate each of these samples by methods from 1(a)-1(c). With this, obtain a Monte Carlo estimator of the mean squared error at each point x of all three estimators (parametric, kernel density estimator with a first order kernel and kernel density estimator with a third order kernel). Plot the (estimated) mean squared errors of all three density estimators as a function of x and comment on the results (provide theoretical justification). (2 points)



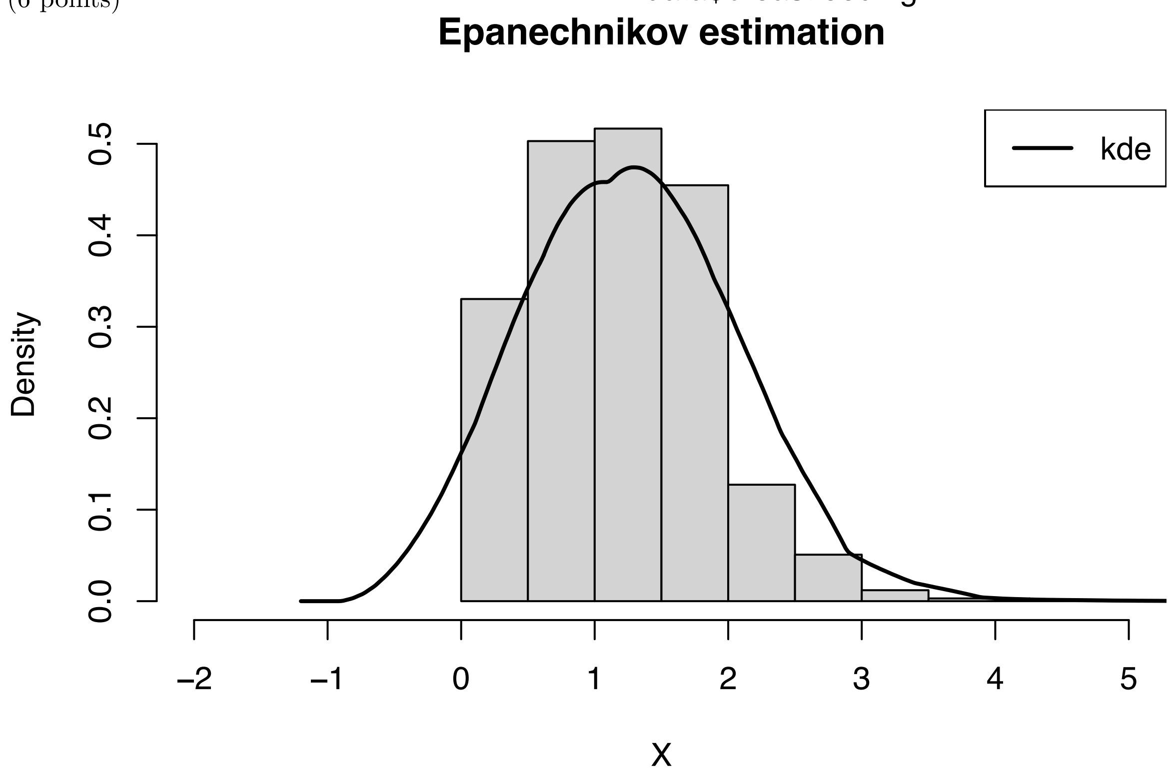
- There is a theorem that shows, that max likelihood estimator, if exists, is an efficient estimator of distribution parameters. Which means, that no estimators can be better. In this case, when sampling from known distribution (gamma), which has max lkelihood estimators for parameters, the best MSE-wise estimator is the parametric one. Order 3 estimator with properly chosen bandwidth also has better estimation then 1st order one, which is shown on graph.

Problem 3

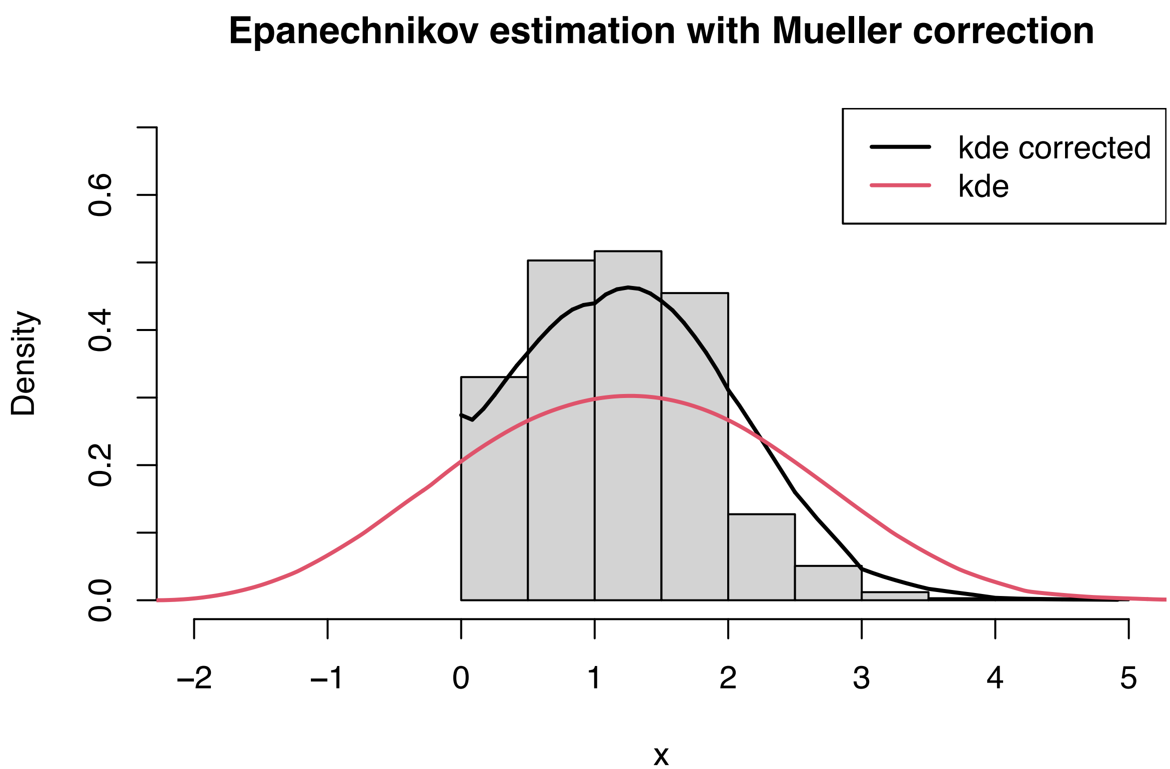
Read the dataset “Kenya DHS” into R and consider the variable breastfeeding, which gives the duration of breastfeeding in months. First, redefine this variable to give the duration in years and plot its histogram. Argue why and at which boundary a correction necessary. Next, estimate the density of this variable employing a kernel density estimator with Epanechnikov kernel and an appropriate boundary correction suggested by Gasser, Mueller. (1979), as given in Lecture 4. Set the bandwidth to h = 0.8. Compare this boundary corrected estimator with a usual kernel density estimator, that uses the same kernel and the same bandwidth, putting both on the histogram. Comment on the results.



(6 points)



Correction is required in x = 0.



- Corrected estimator correctly deals with 0 point. With this bandwidth corrected estimator deas better not only with 0, but also with density itself.

```r
# Problem 2
# 1. Simulate a dataset of 500 observations from the Gamma distribution
# with the shape pa- rameter 10 and the rate parameter 1.
# Set seed to 2425 to ensure comparability.
set.seed(2425)
alpha <- 10
betta <- 1
gamma_data <- rgamma(500, shape = alpha, rate = betta)
# (a) Obtain a parametric estimator for the density of these data,
# employing moment estimators for the parameters. (1 point)

# Estimators
# alpha := Mean(X, I)^2/Variance(X, I), beta := Variance(X, I)/Mean(X, I)

sample_mean <- mean(gamma_data)
sample_variance <- var(gamma_data)
alpha_hat <- (sample_mean^2) / sample_variance
beta_hat <- sample_mean / sample_variance

hist(gamma_data, freq=FALSE, ylim=c(0, 0.2),
     main = "Histogram with parametric estimation",)
curve(dgamma(x, shape = alpha_hat, rate = beta_hat), add = TRUE, col = 2, lwd = 2)
legend("topright", legend = "parametric estimation",
       col = 2, lwd = 2)

# (b) Obtain a kernel density estimator with Epanechnikov kernel and a
# bandwidth chosen by cross-validation. Ensure that the cross-validation
# criterion has a global minimum by plotting it on a suitable range of
# bandwidths. (2 points)

# common functions
#kernel density estimator
kde <- function(x,h,ker) apply(ker(outer(x,x,"-")/h),2,mean)/h
#cross-validation
CV<-function(h,X,kern)
{
  n=length(X)
  K=kern(outer(X,X,"-")/h)
  diag(K)=rep(0,n)
  integrand<-function(x) (mean(kern((X-rep(x,n))/h))/h)^2
  CV=integrate(Vectorize(integrand),lower=-Inf,upper=Inf,subdivisions=2000,
rel.tol=1e-3)$value-2*sum(K)/(n*(n-1)*h)
}


# Kernel
kernel1 <- function(x) ifelse(abs(x) <= 1, 0.75 * (1 - x^2), 0)
plot(kernel1, -2, 2, main = "Epanechnikov kernel function")
grid=100
X=sort(gamma_data)
h1.grid=seq(0.01,6,length=grid)
cv=rep(NA,grid)
cv=Vectorize(CV,vectorize.args=c("h"))(h1.grid,X,kernel1)
#find minimum
h1.cv=optimize(CV,interval=c(1,10),X=X,kern=kernel1)$minimum
print(h1.cv)
#plot kde with h.cv and h.Jh
plot(X,kde(X,h=h1.cv,ker=kernel1),lwd=2,type="l",ylim=c(0,0.2),xlab="x",ylab="KDE")
rug(X)
lines(X,dgamma(X, alpha, betta),lwd=2,col=2)
legend("topleft",col=c(1,2),lty=1,c("CV","true"),lwd=2)

print(cat("bandwidth", h1.cv))

?cat

#(c) Obtain a kernel density estimator as in (b), replacing Epanechnikov kernel by
a kernel
```

```
#of order 3 (1 point)

# Kernel
kernel2 <- function(x) ifelse(abs(x) <= 1, 45/32*(1 - 10/3*x^2 + 7/3*x^4), 0)
plot(kernel2, -2, 2, main = "3-rd order kernel function")
grid=100
X=sort(gamma_data)
h2.grid=seq(0.01,6,length=grid)
cv=rep(NA,grid)
cv=Vectorize(CV,vectorize.args=c("h"))(h2.grid,X,kernel2)
#find minimum
h2.cv=optimize(CV,interval=c(1,10),X=X,kern=kernel2)$minimum
print(h2.cv)
#plot kde with h.cv and h.Jh
plot(X,kde(X,h=h2.cv,ker=kernel2),lwd=2,type="l",ylim=c(0,0.2),xlab="x",ylab="KDE")
rug(X)
lines(X,dgamma(X, alpha, betta),lwd=2,col=2)
legend("topleft",col=c(1,2),lty=1,c("CV","true"),lwd=2)


#Plot estimators obtained in (a) - (c) together with the true density on one plot,
#putting a legend. Comment on the results. Compare also the bandwidths obtained in (b) and (c):
#which one is larger and why (give theoretical justification)? (2 points)

plot(X,kde(X,h=h1.cv,ker=kernel1),lwd=2,type="l",ylim=c(0,0.2),xlab="x",ylab="KDE", col=1)
lines(X,kde(X,h=h2.cv,ker=kernel2),lwd=2, col=2)
rug(X)
lines(X,dgamma(X, alpha, betta),lwd=2,col=3)
legend("topright",col=c(1, 2, 3),c("epanechnikov", "3rd order", "true"),lwd=2)
h1.cv
h2.cv



set.seed(2425)
n_sim <- 300
n_samples <- 500
alpha <- 10
betta <- 1

mse_parametric <- numeric(length(n_samples))
mse_epanechnikov <- numeric(length(n_samples))
mse_order3 <- numeric(length(n_samples))

for (i in 1:n_sim) {

  sample_data <- sort(rgamma(n_samples, alpha, betta))
  x_vals = seq(0, max(sample_data), length.out = n_samples)

  sample_mean <- mean(sample_data)
  sample_var <- var(sample_data)
  alpha_hat <- sample_mean^2 / sample_var
  beta_hat <- sample_mean / sample_var
  parametric_density <- dgamma(sample_data, shape = alpha_hat, rate = beta_hat)

  kde_epanechnikov <- kde(sample_data,h=h1.cv,ker=kernel1)

  kde_order3 <- kde(sample_data,h=h2.cv,ker=kernel2)

  true_density <- dgamma(sample_data, shape = alpha, rate = betta)

  mse_parametric <- mse_parametric + (parametric_density - true_density)^2
  mse_epanechnikov <- mse_epanechnikov + (kde_epanechnikov - true_density)^2
  mse_order3 <- mse_order3 + (kde_order3 - true_density)^2
}

mse_parametric <- mse_parametric / n_sim
mse_epanechnikov <- mse_epanechnikov / n_sim
```

```r
mse_order3 <- mse_order3 / n_sim

plot(x_vals, mse_parametric, type = "l", col = "blue", lwd = 2, ylim=c(0, 1e-4),
     xlab = "x", ylab = "Mean Squared Error (MSE)", main = "MSE of Density Estimators")
lines(x_vals, mse_epanechnikov, col = "red", lwd = 2)
lines(x_vals, mse_order3, col = "green", lwd = 2)
legend("topright", legend = c("Parametric", "Epanechnikov KDE", "Order-3 KDE"),
       col = c("blue", "red", "green"), lwd = 2)




# Problem 3
#Read the dataset “Kenya DHS” into R and consider the variable breastfeeding,
#which gives the duration of breastfeeding in months.
#First, redefine this variable to give the duration in years and plot its histogram.
#Argue why and at which boundary a correction necessary.
#Next, estimate the density of this variable employing a kernel density estimator
#with Epanechnikov kernel and an appropriate boundary correction suggested
#by Gasser, T. and Mu¨ller, H.G. (1979), as given in Lecture 4.
#Set the bandwidth to h = 0.8. Compare this boundary corrected estimator
#with a usual kernel density estimator, that uses the same kernel and the
#same bandwidth, putting both on the histogram. Comment on the results. (6 points)

?density

setwd("/Users/khodosevichleo/Desktop/HauptUni/1sem/CSM/Excercises/")
data=read.table("KenyaDHS.txt",header=TRUE)
attach(data)
head(data)

data$breastfeeding <- data$breastfeeding / 12
hist(data$breastfeeding)
# We need to use boundary correction at point 2, as there will be a huge drop in density function
X <- sort(data$breastfeeding)
# common functions
#kernel density estimator
kde <- function(x,h,ker) apply(ker(outer(x,x,"-")/h),2,mean)/h
kernel1 <- function(x) ifelse(abs(x) <= 1, 0.75 * (1 - x^2), 0)
#cross-validation
h = 1
hist(X, freq=FALSE)
lines(X, kde(X, h, kernel1))
#lines(density(X,kernel="epanechnikov"),lwd=2,type="l", xlab="x",ylab="KDE")
legend("topright",col=c(1),lty=1,c("kde"),lwd=2)



ker <- kernel1
moments <- function(alpha,degree)
{
  integrand <- function(x) {x^degree*ker(x)}
  integrate(integrand,-1,alpha)
}
x <- seq(-1,1,length=300)
bKernel <- function(alpha,x)
{
  (moments(alpha,2)$value-moments(alpha,1)$value*x)*ker(x)*(x<=alpha&x>=-1)/(moments(alpha,0)$value*moments(alpha,2)$value-(moments(alpha,1)$value)^2)
}

x <- de <- sort(data$breastfeeding)
h=2

for(i in 1:length(x))
{
```

```
    if (x[i]>h)
      de[i] <- sum(ker((x[i]-x)/h))/(h*n)
    else
    {
      alpha <- x[i]/h
      de[i] <- sum(bKernel(alpha,(x[i]-x)/h))/(h*n)
    }
  }

hist(x, freq=FALSE, ylim=c(0, 0.5))
lines(x, de, lwd=2, xlab="x",ylab="KDE")
legend("topright",col=c(1),lty=1,c("kde"),lwd=2)
```