

############## 3.1 a ################

a0 <- function(x){return ((x/(1 + (x)^2)) \* exp(-(-x)^2/2))}

a2 <- function(x){return ((x/(1 + (x)^2)) \* exp(-(2-x)^2/2))}

a4 <- function(x){return ((x/(1 + (x)^2)) \* exp(-(4-x)^2/2))}

b0 <- function(x){return ((1/(1 + (x)^2)) \* exp(-(-x)^2/2))}

b2 <- function(x){return ((1/(1 + (x)^2)) \* exp(-(2-x)^2/2))}

b4 <- function(x){return ((1/(1 + (x)^2)) \* exp(-(4-x)^2/2))}

par(mfrow = c(3,1))

plot(a0,xlim=c(-3,3),ylim=c(-1,1), main = "X = 0")

plot(b0,xlim=c(-3,3),add=TRUE,ylim=c(-1,1))

plot(a2,xlim=c(-3,5),ylim=c(-0.5,0.5),main = "X = 2")

plot(b2,xlim=c(-3,5),add=TRUE,ylim=c(-0.5,0.5))

plot(a4,xlim=c(-3,7),ylim=c(-0.25,0.25),main = "X = 4")

plot(b4,xlim=c(-3,7),add=TRUE,ylim=c(-0.25,0.25))

par(mfrow = c(1,1))

plot(rcauchy(10^4))

set.seed(123)

par(mfrow = c(3,3))

for(r in c(0,2,4)){

Nsim <- 10^4

cauch <- rcauchy(Nsim)

cumsum\_num <- cumsum(cauch \* dnorm(cauch,mean = r))/(1:10^4)

cumsum\_denom <- cumsum(dnorm(cauch,mean = r))/(1:10^4)

plot(cumsum\_num,type = "l",main = paste("numerator with x",r))

plot(cumsum\_denom, type = "l",main = paste("denominator with x",r))

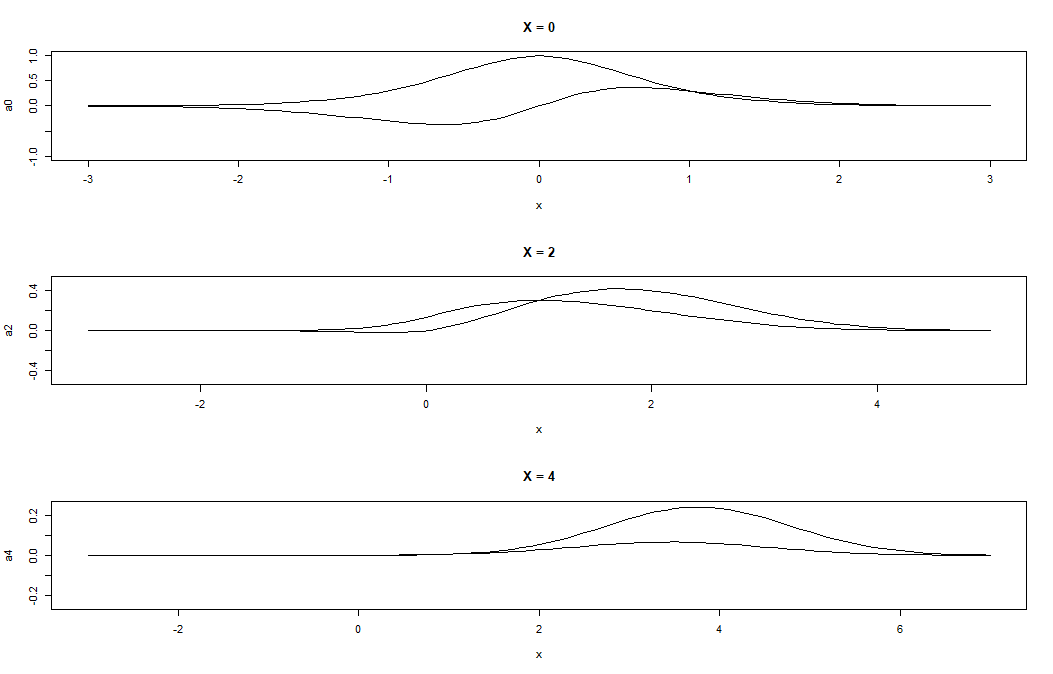
print(cumsum\_num[length(cumsum\_num)]/cumsum\_denom[length(cumsum\_denom)])

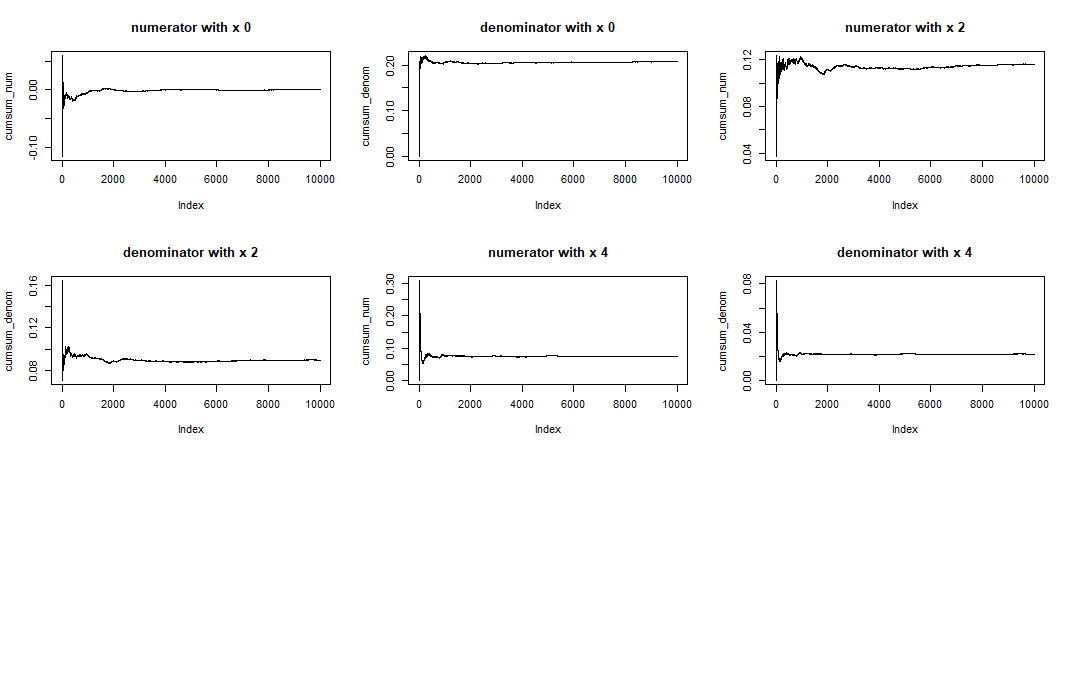
}

# [1] 0.0006678213

# [1] 1.276915

# [1] 3.408064





###### 3.1 b ######

par(mfrow = c(3,3))

for(r in c(0,2,4)){

Nsim <- 10^4

cauch <- rcauchy(Nsim)

cumsum\_num\_err <- sqrt(cumsum((cauch \* dnorm(cauch,mean = r))^2))/(1:10^4)

cumsum\_denom\_err <- (cumsum((dnorm(cauch,mean = r))^2))/(1:10^4)

plot(cumsum\_num\_err,type = "l",main = paste("numerator with x",r))

plot(cumsum\_denom\_err, type = "l",main = paste("denominator with x",r))

print(paste("standerror with 0.95 probabilty for x value",r,"is",

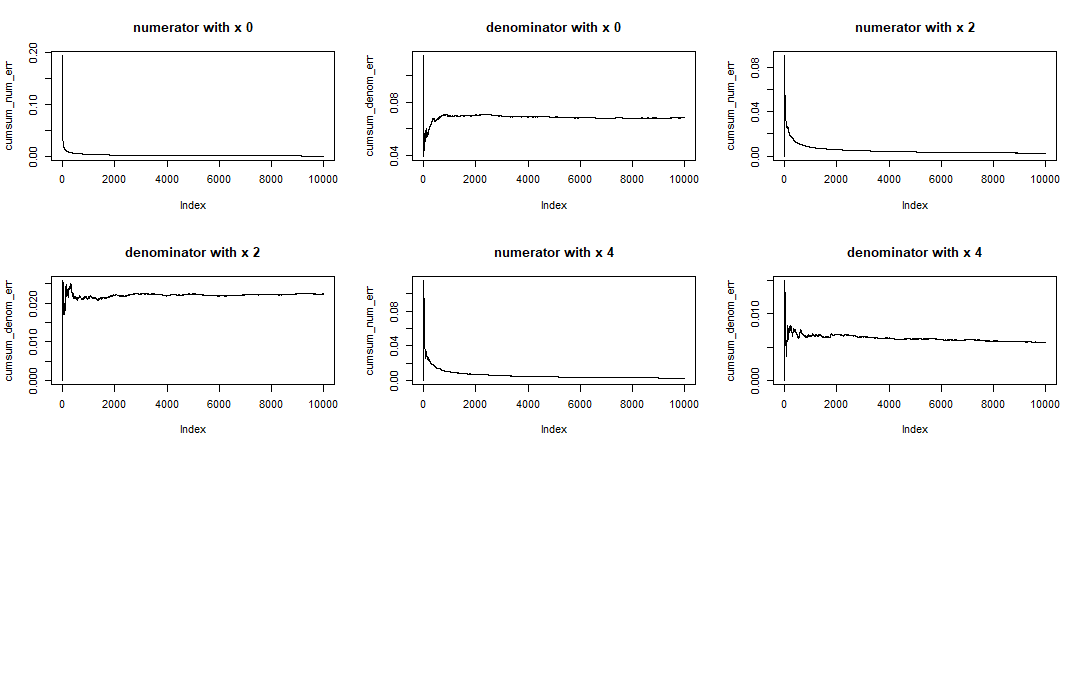
+cumsum\_num\_err[length(cumsum\_num\_err)]/cumsum\_denom\_err[length(cumsum\_denom\_err)]))

}

# [1] "standerror with 0.95 probabilty for x value 0 is 0.0215398394549276"

# [1] "standerror with 0.95 probabilty for x value 2 is 0.115831689937375"

# [1] "standerror with 0.95 probabilty for x value 4 is 0.488633976461582"



##### 3.1 c #######

set.seed(123)

par(mfrow = c(3,3))

for(r in c(0,2,4)){

Nsim <- 10^4

#cauch <- rcauchy(Nsim)

cauch <- rnorm(Nsim,mean = r)

cumsum\_num <- cumsum(cauch \* dcauchy(cauch))/(1:10^4)

cumsum\_denom <- cumsum(dcauchy(cauch))/(1:10^4)

plot(cumsum\_num,type = "l",main = paste("numerator with x",r))

plot(cumsum\_denom, type = "l",main = paste("denominator with x",r))

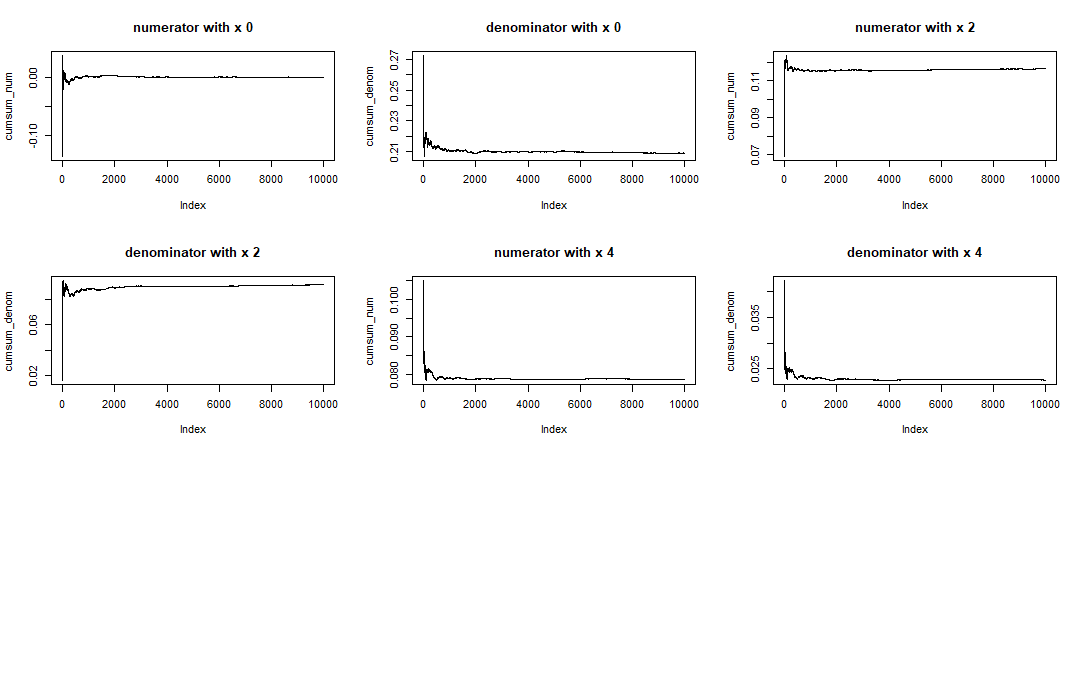
print(cumsum\_num[length(cumsum\_num)]/cumsum\_denom[length(cumsum\_denom)])

}

# [1] -0.001183145

# [1] 1.273159

# [1] 3.43772



Niter=10^4

co=rcauchy(Niter)

I=mean(co\*dcauchy(co,mean=x))/mean(dnorm(co,mean=x))

#We thus get

x=0

mean(co\*dcauchy(co,mean=x))/mean(dnorm(co,mean=x))

#0.01724

x=2

mean(co\*dnorm(co,mean=x))/mean(dnorm(co,mean=x))

#[1] 1.295652

x=4

mean(co\*dnorm(co,mean=x))/mean(dnorm(co,mean=x))

#[1] 3.107256

> ##################### 3.3 ##########################

> par(mfrow=c(1,1))

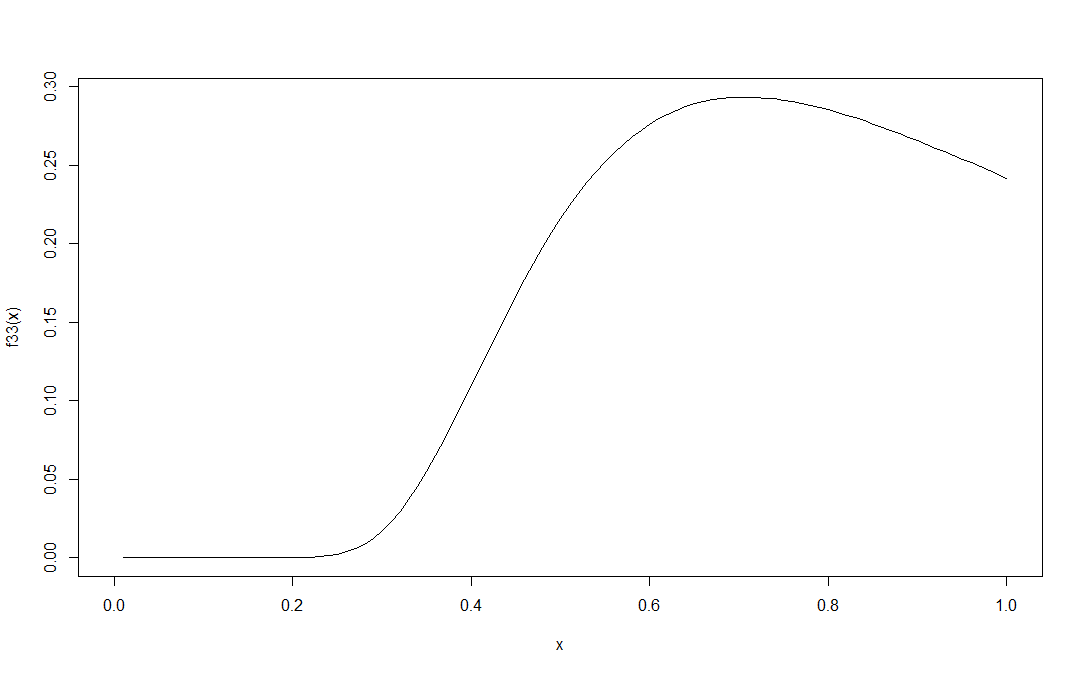
> f33 <- function(x){

+ return (1/(x\*x\*sqrt(2\*pi)\*exp(1/(2\*x\*x))))

+ }

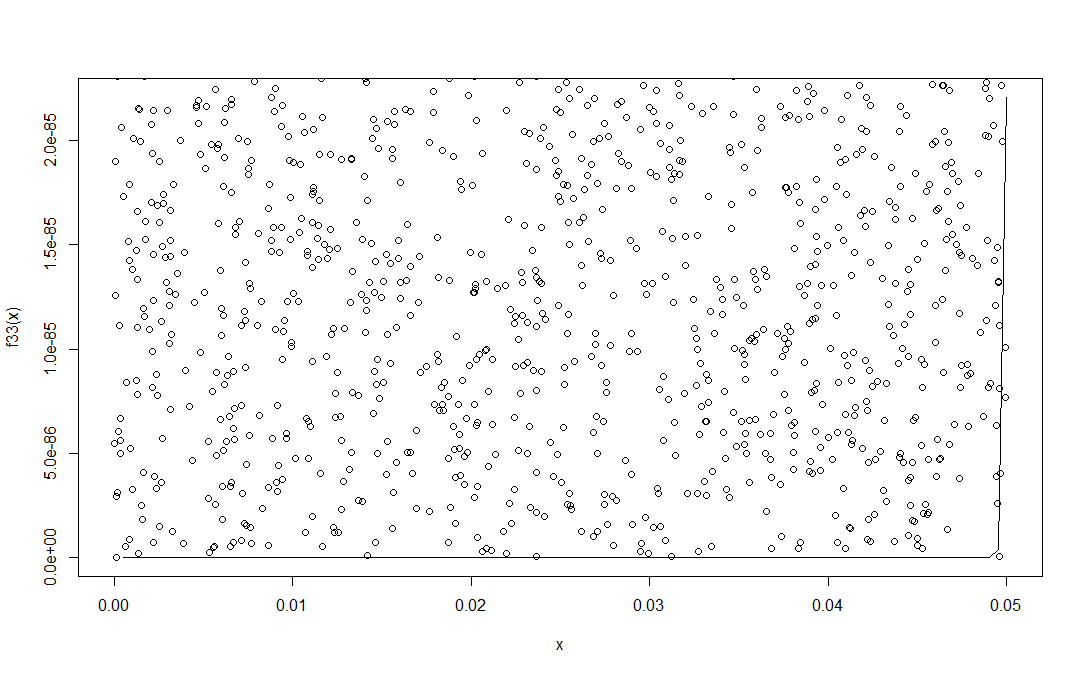
>

> curve(f33,from = 0, to = 1)



> curve(f33,from = 0, to = 1/20)

> points(runif(1000,0,0.05),runif(1000,0,2.5e-85))



> X <- runif(10^4,0,1/20)

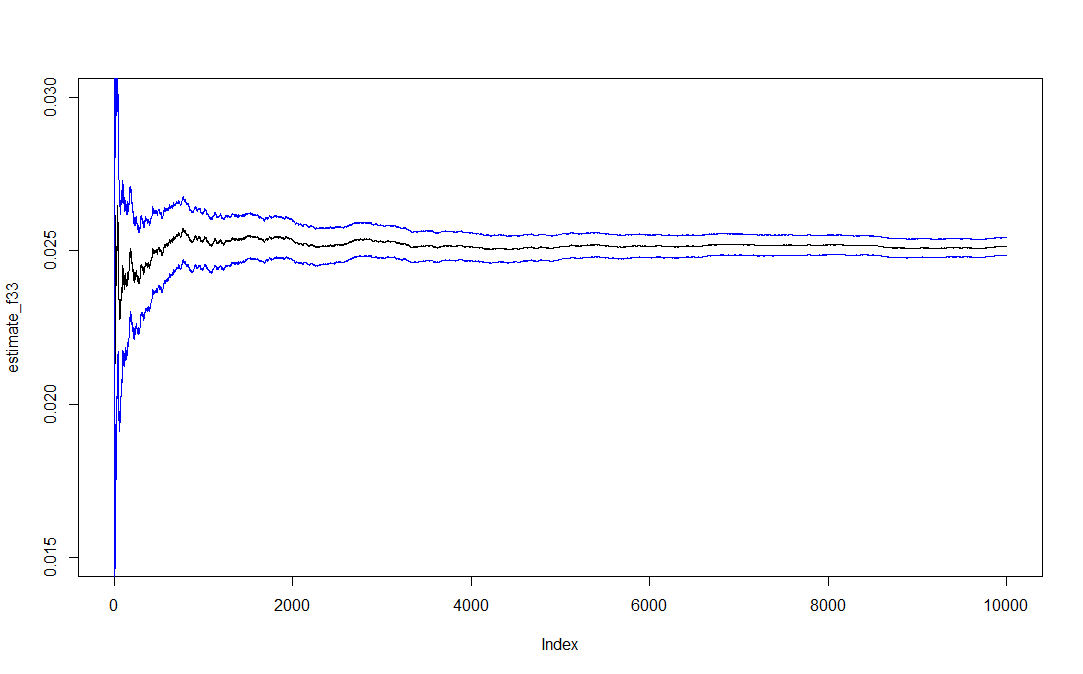
> estimate\_f33 <- cumsum(X) / (1:10^4)

> esterr\_f33 <- sqrt(cumsum((X - estimate\_f33)^2))/(1:10^4)

> plot(estimate\_f33,type = "l",ylim = c(0.015,0.030))

> lines(estimate\_f33 + 2 \* esterr\_f33, col = "blue")

> lines(estimate\_f33 - 2 \* esterr\_f33, col = "blue")



> estimate\_f33[length(estimate\_f33)]

[1] 0.02514024

> (estimate\_f33 + 2 \* esterr\_f33)[10^4]

[1] 0.02542779

> (estimate\_f33 - 2 \* esterr\_f33)[10^4]

[1] 0.02485269

> #Using Integrate function

> integrate(f33,0,1/20)

2.759158e-89 with absolute error < 5.4e-89

>

> #Using accept reject method

> set.seed(12)

> ptunder <- sum(f33(runif(10^6,0,1/20)) > runif(10^6,0,2.5e-85))

> (ptunder / 10^6) \* 0.05 \* 2.56e-85

[1] 2.77888e-89

> ############## 3.4 b ################

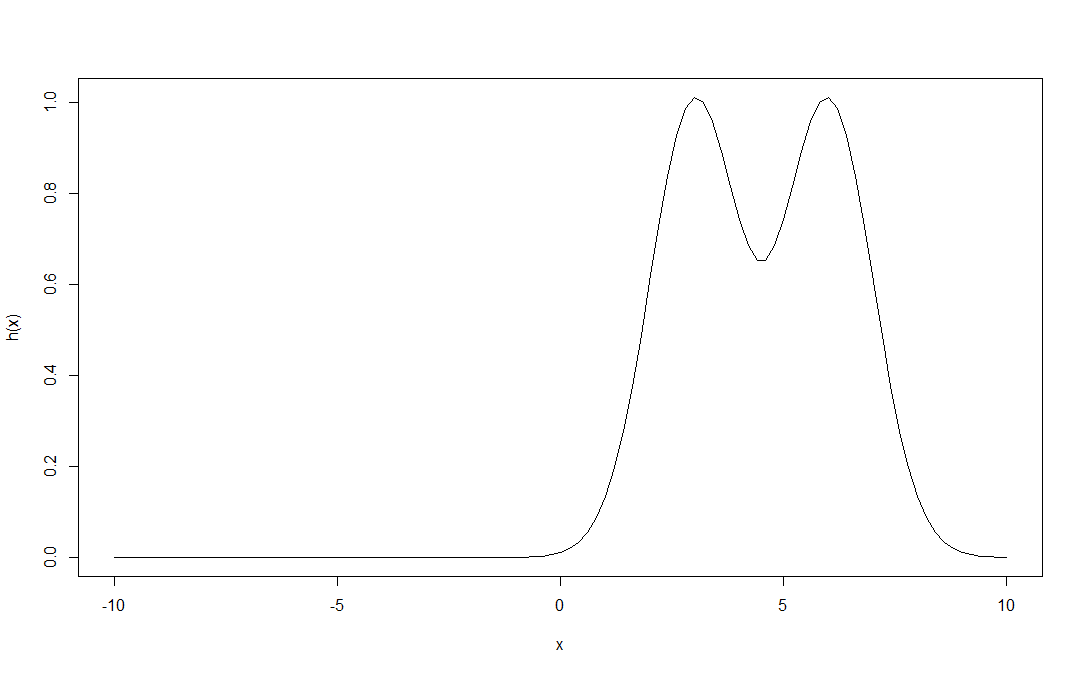
>

> h <- function(x){

+ return (exp((-(x-3)^2)/2) + exp((-(x-6)^2)/2))

+ }

> curve(h, from = -10, to = 10)

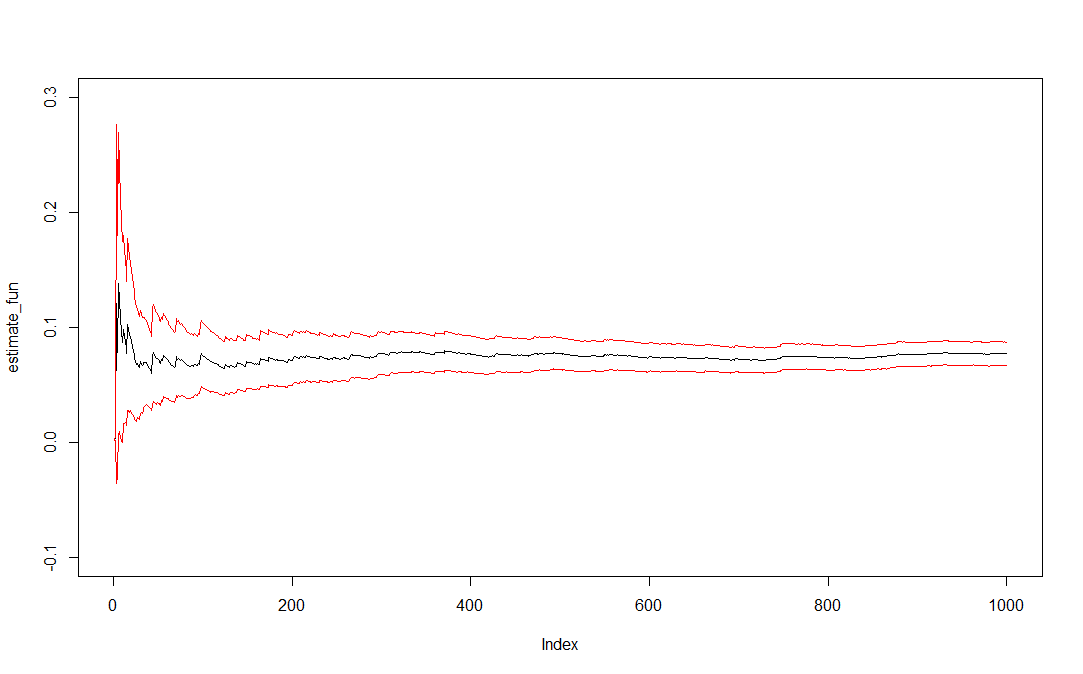


> set.seed(123)

> x <- h(rnorm(10^3))

> estimate\_fun <- cumsum(x) /(1:10^3)

> plot(estimate\_fun,type="l",ylim = c(-0.1,0.3))



> estimate\_fun[length(estimate\_fun)]

[1] **0.07699182**

> esterr <- sqrt(cumsum((x-estimate\_fun)^2))/(1:10^3)

> lines(estimate\_fun + 2\*esterr, col = "red")

> lines(estimate\_fun - 2\*esterr, col = "red")

>

> (estimate\_fun + 2\*esterr)[1000]

[1] 0.08712949

> (estimate\_fun - 2\*esterr)[1000]

[1] 0.06685415

>

> (exp(-9/4)+exp(-9))/sqrt(2)

[1] **0.07461577**