

统计软件 HW10

邵智轩

1400012141

物理学院

核回归

```
tri_cube<-function(t){
  abs.t=abs(t)
  ifelse(abs.t>1,0,(1-abs.t^3)^3)
}

kernelEstimate<-function(x0,x,y,kernel=tri_cube,
                          h=1.06*sd(x)*length(x)^(-1/5)){
  K<-kernel((x0-x)/h)
  sum(K*y)/sum(K)
}

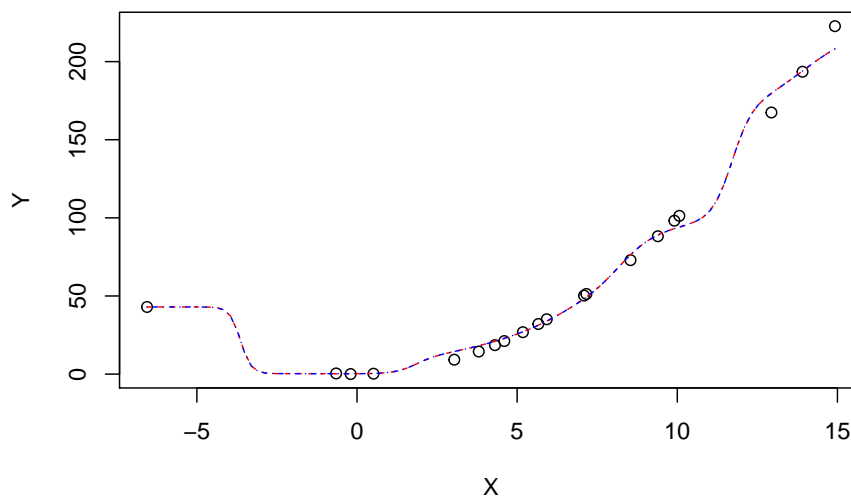
kernelSmoother<-function(x,y,kernel=tri_cube,
                          n.points=max(100L,length(x)),
                          h=1.06*sd(x)*length(x)^(-1/5),
                          x.range=range(x),
                          x.points=seq(from=x.range[1],to=x.range[2],
                                         length.out = n.points)){
  y.points<-vapply(x.points,
                   function(x0) kernelEstimate(x0,x,y,kernel = kernel,h = h),
                   FUN.VALUE = 0.5)
  return(cbind(x.points,y.points))
}

# 数据模拟测试
```

```

set.seed(2)
X<-rnorm(20,5,5)
Y<-X^2
plot(X,Y)
# 先看看和 R 标准函数的拟合结果是否一致
points(ksmooth(X,Y,kernel = "normal",bandwidth = qnorm(0.75)/0.25),
       type="l",lty=2,col="red")
points(kernelSmoother(X,Y,kernel = dnorm,h = 1),type="l",lty=4,col="blue")

```

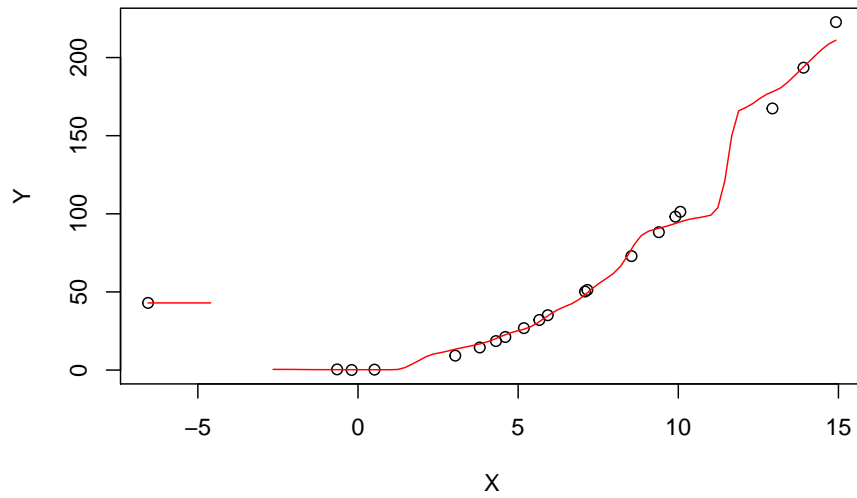


当 bandwidth h 调成相同时，拟合结果是一致的，两条曲线重合在一起。

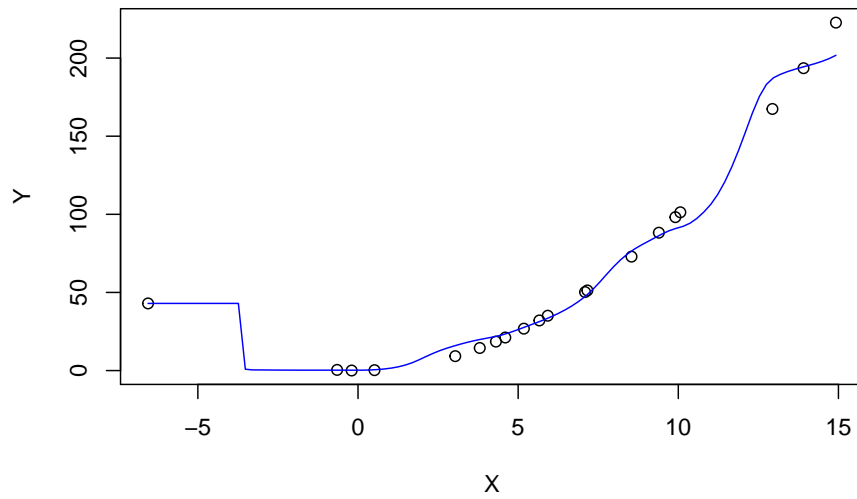
```

plot(X,Y)
points(kernelSmoother(X,Y,h = 2),type="l",col="red")

```



```
plot(X,Y)
points(kernelSmoother(X,Y),type="l",col="blue") # default width h
```

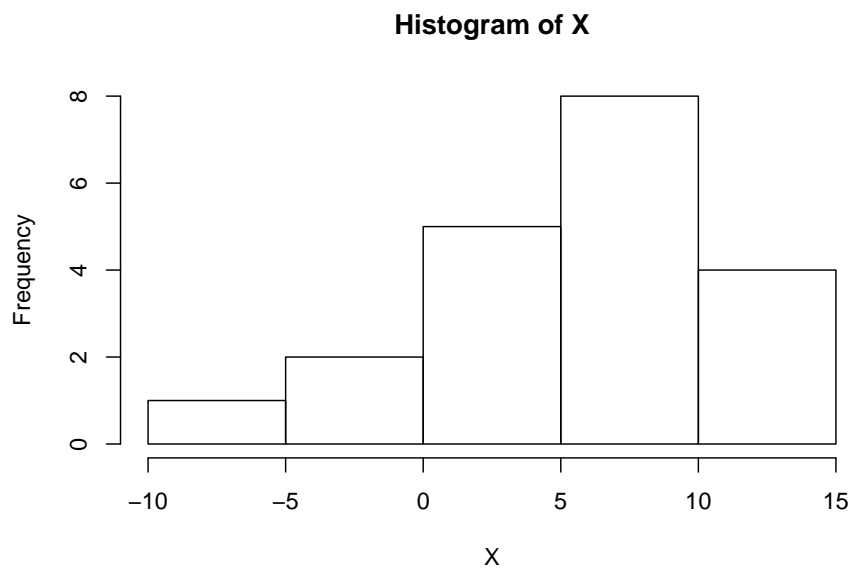



```

        x.range=range(x),
        cut=3,
        x.points=seq(from=x.range[1]-cut*h,to=x.range[2]+cut*h,
                      length.out = n.points)){
f.points<-vapply(x.points,
                 function(x0) kernelDensityEstimate(x0,x,kernel = kernel,h = h),
                 FUN.VALUE = 0.5)
return(cbind(x.points,f.points))
}

hist(X)

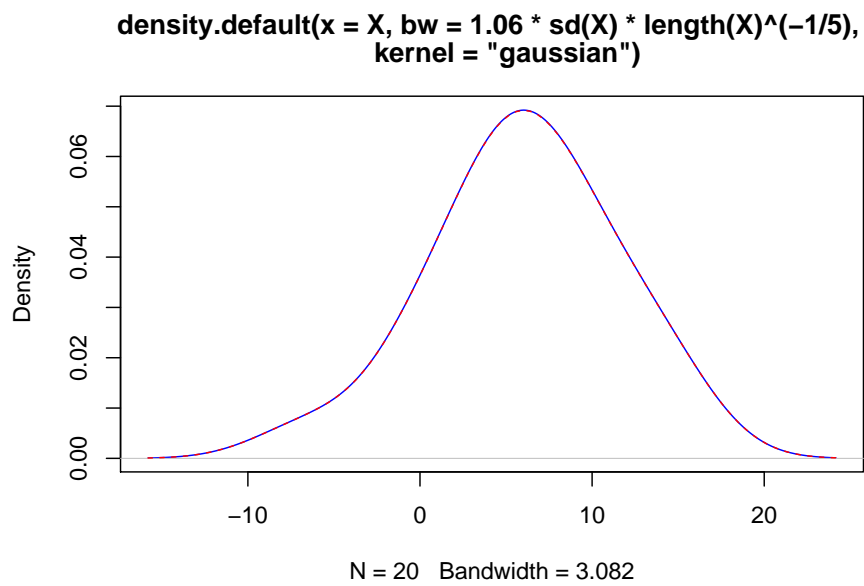
```



```

plot(density(X,bw=1.06*sd(X)*length(X)^(-1/5),kernel = "gaussian"),col="blue")
points(kernelDensity(X),type='l',col="red",lty=2)

```



两条曲线是重合的，给出的结果与 R 标准函数 `density` 一致。