统计咨询与实践第三次作业

张申铎 2176112379, 祖劭康 2173412124

目录

1 问题 1

使用核函数估计密度函数，基本模型如下

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K(\frac{x - x_i}{h})$$

其中在本次实验中使用的核函数为矩形和高斯核，形式如下

矩形

$$K(x) = \begin{cases} \frac{1}{2} & if\ |x| \leq 1 \\ 0 & otherwise \end{cases}$$

高斯核

$$K(x) = \frac{1}{\sqrt{2\pi}} exp(-\frac{1}{2}x^2)$$

1.1 数据生成

生成来自 $N(2,1)$ 和 $\Gamma(3,2)$ 的样本各 200 个，准备对其密度函数做估计

```
data1 <- rnorm(200,2,1)
data2 <- rgamma(200, 3 , 2)
```

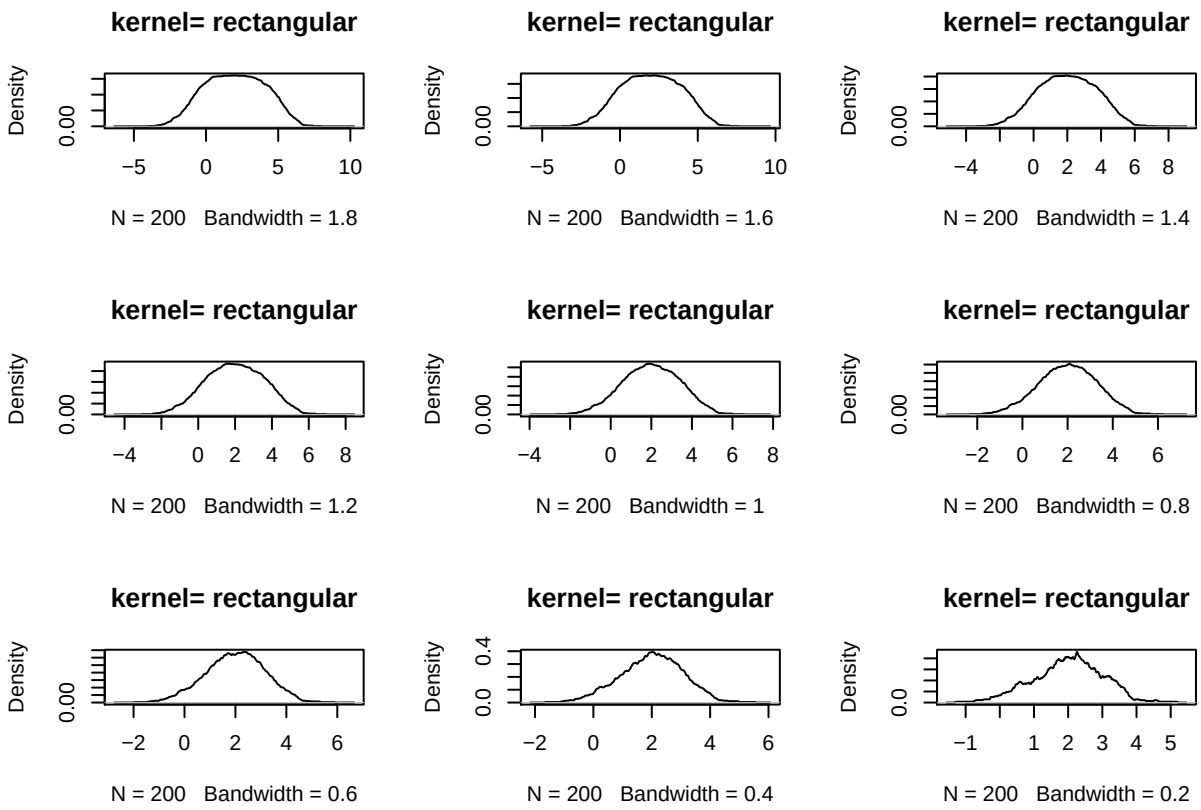
1.2 模拟结果

分别对来自两个总体的数据使用分别使用高斯核和矩形估计密度函数，并设置从 1.8 到 0.2 的间隔为 0.2 的窗宽，观察窗宽变化及核函数的不同对估计结果的影响

1.2.1 来自 $N(2,1)$ 的样本

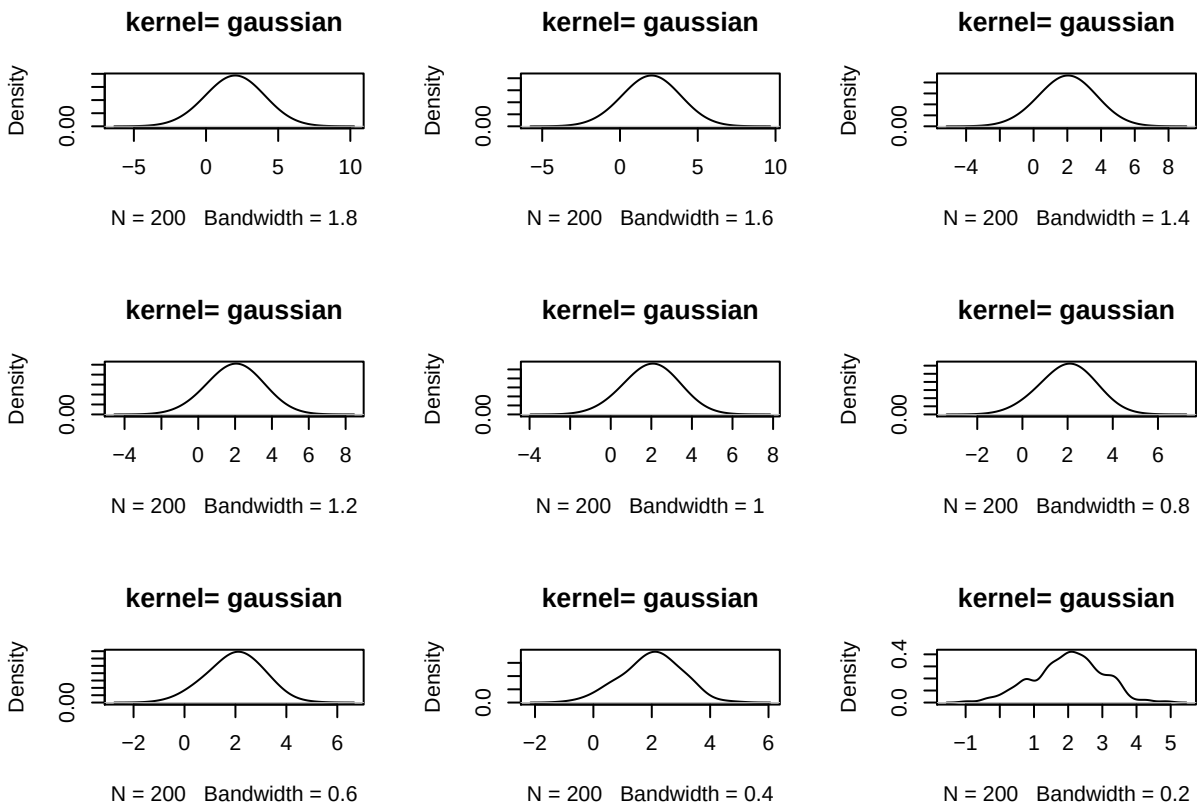
1.2.1.1 核函数使用矩形

```
block <- 1.8
par(mfrow=c(3,3))
for(i in 1: 9){
    plot(density(data1,bw = block,kernel= "rectangular"),main = "kernel= rectangular")
    block <- block - 0.2
}
```



1.2.1.2 核函数使用高斯

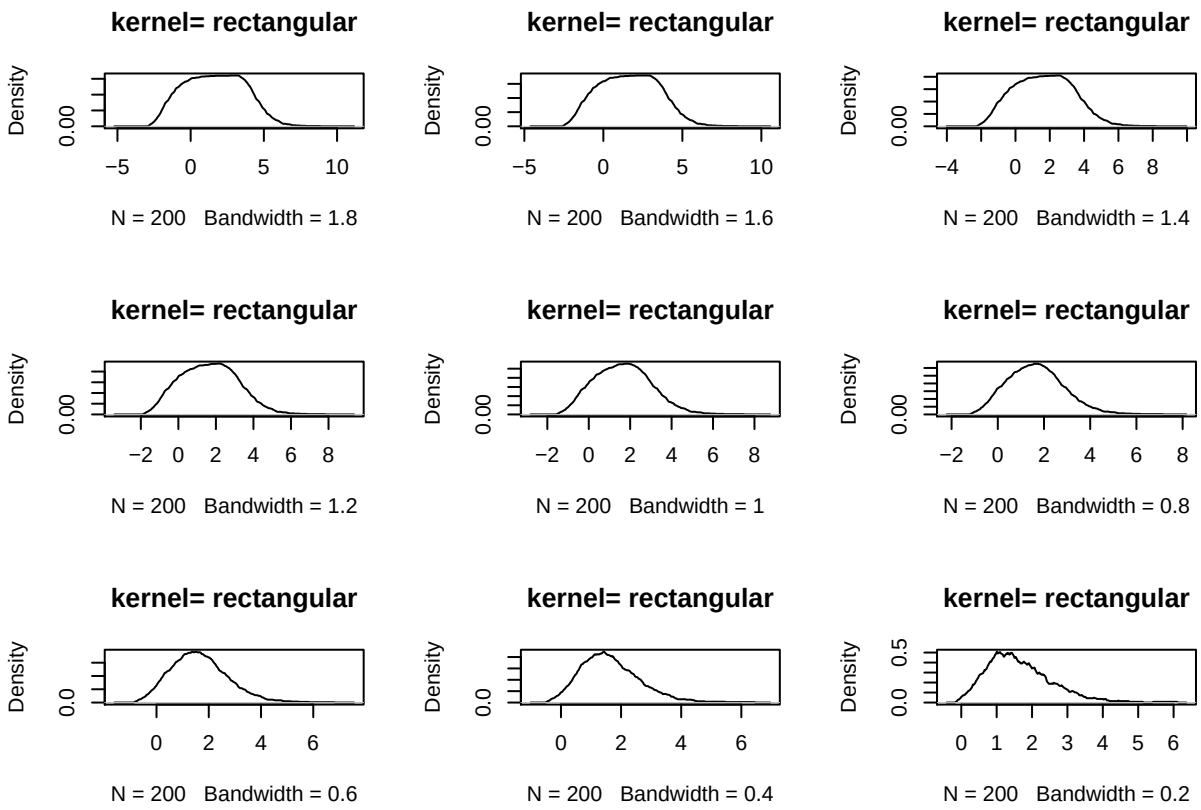
```
block <- 1.8
par(mfrow=c(3,3))
for(i in 1: 9){
    plot(density(data1,bw = block,kernel= "gaussian"),main = "kernel= gaussian")
    block <- block - 0.2
}
```



1.2.2 来自 $\Gamma(3,2)$ 的样本

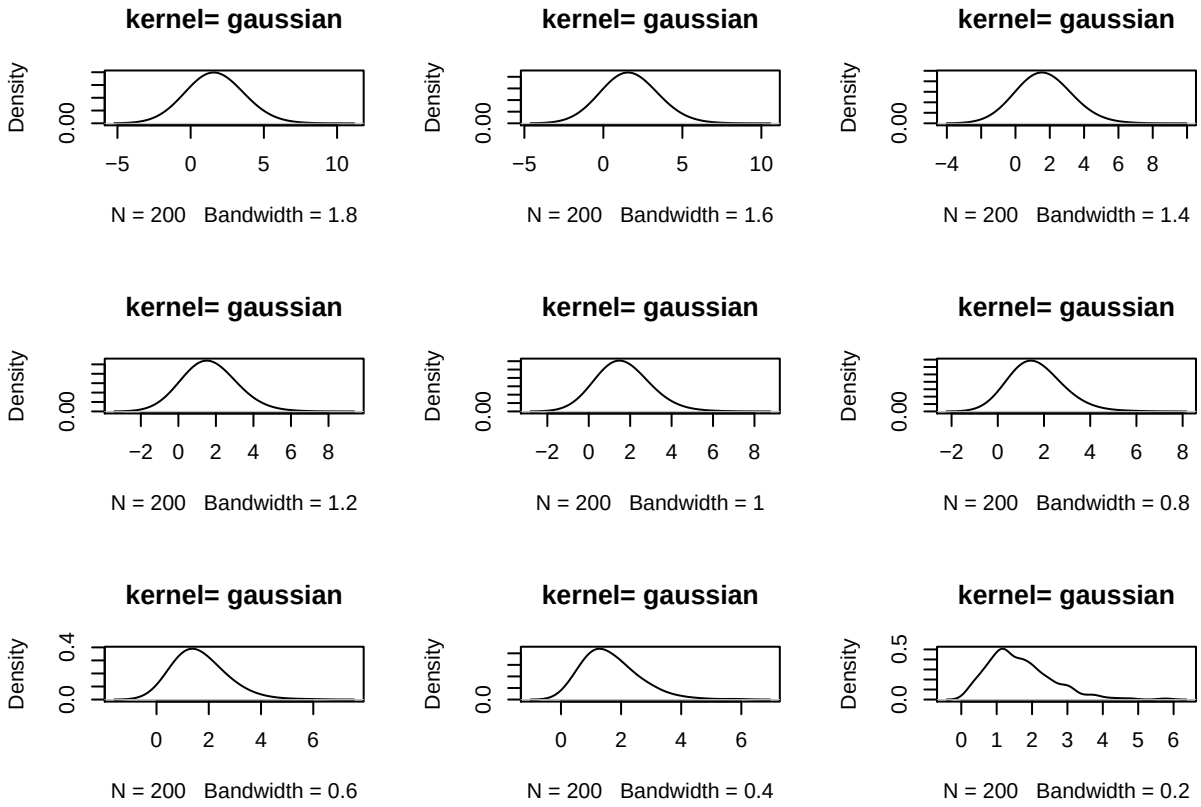
1.2.2.1 核函数使用矩形

```
block <- 1.8
par(mfrow=c(3,3))
for(i in 1: 9){
    plot(density(data2,bw = block,kernel= "rectangular"),main = "kernel= rectangular")
    block <- block - 0.2
}
```



1.2.2.2 核函数使用高斯

```
block <- 1.8
par(mfrow=c(3,3))
for(i in 1: 9){
    plot(density(data2,bw = block,kernel= "gaussian"),main = "kernel= gaussian")
    block <- block - 0.2
}
```



1.3 总结

有实验结果的图片可以看出，窗宽的选择对估计结果影响较大。如果窗宽选择过大，会导致欠拟合，如果窗宽设置过小，则会过拟合。核的影响在这次采用简单数据的实验中并不明显，但也可明显看出不同核函数拟合出的函数形态有明显区别。由核函数的性质，很自然的使用矩形拟合的曲线光滑度不如高斯核。

2 问题 2

我们考虑一个非参数回归问题

$$Y_i = f(X_i) + \epsilon_i, i = 1, 2 \ldots \tag{1}$$

$$f(x) = \log(x+1) - x^2 + 5\left(x - \frac{1}{2}\right)^5 \tag{2}$$

我们使用核方法来对回归函数 f 进行非参数估计，我们对比了不同种类 Kernel 的选择对估计的影响。

```
library(np)
```

```
## Warning: package 'np' was built under R version 4.0.3
```

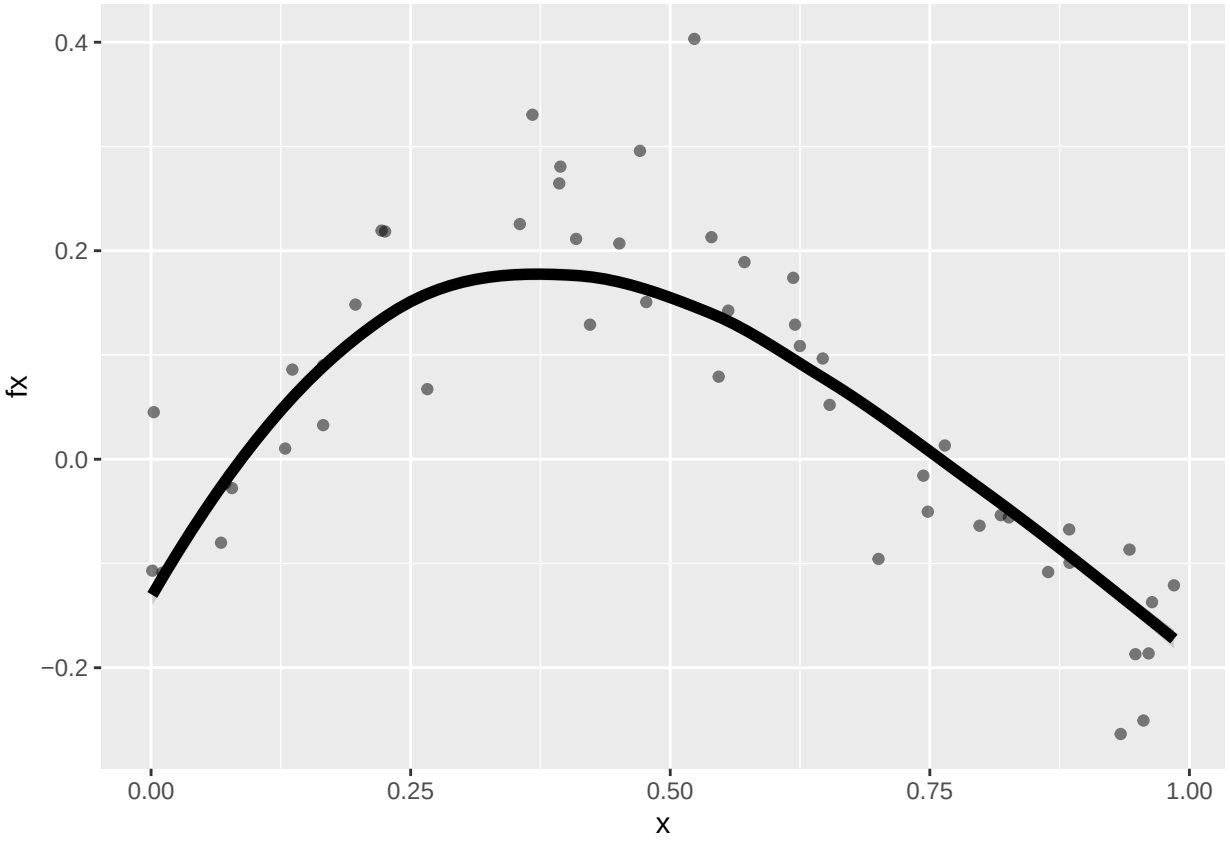
```
library(ggplot2)
set.seed(2020)
```

2.1 生成数据

```
samplesize <- 50
x <- runif(samplesize)
fx <- log(x+1)- x^2+5 * (x-1/2)^5
y <- fx + rnorm(samplesize,mean=0,sd =0.08)
data<- data.frame(x=x,y=y,fx=fx)
```

我们对目标函数与样本数据进行一次可视化

```
ggplot(data = data)+geom_smooth(aes(x=x,y=fx),color="black",size=2)+geom_jitter(aes(x=x,y=y),alpha
```



2.2 N-W estimator

我们采用 N-W estimator 对回归函数进行非参数估计，考虑两种核函数选择，一种 Epanechnikov 核函数，一种 Gaussian 核函数。通过几次实验我们选择了估计效果最好的核函数的 2 阶版本。其中带宽基于 cross validation 自动选择。

2.2.1 Gaussian Kernel

首先我们采用高斯核函数进行估计，高斯核函数具有形式

$$K(u) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{u^2}{2}\right) \tag{3}$$

```
model.1 <- npreg(data$y ~ data$x,bwmethod="cv.aic",ckertype ='gaussian',ckerorder = 2)
```

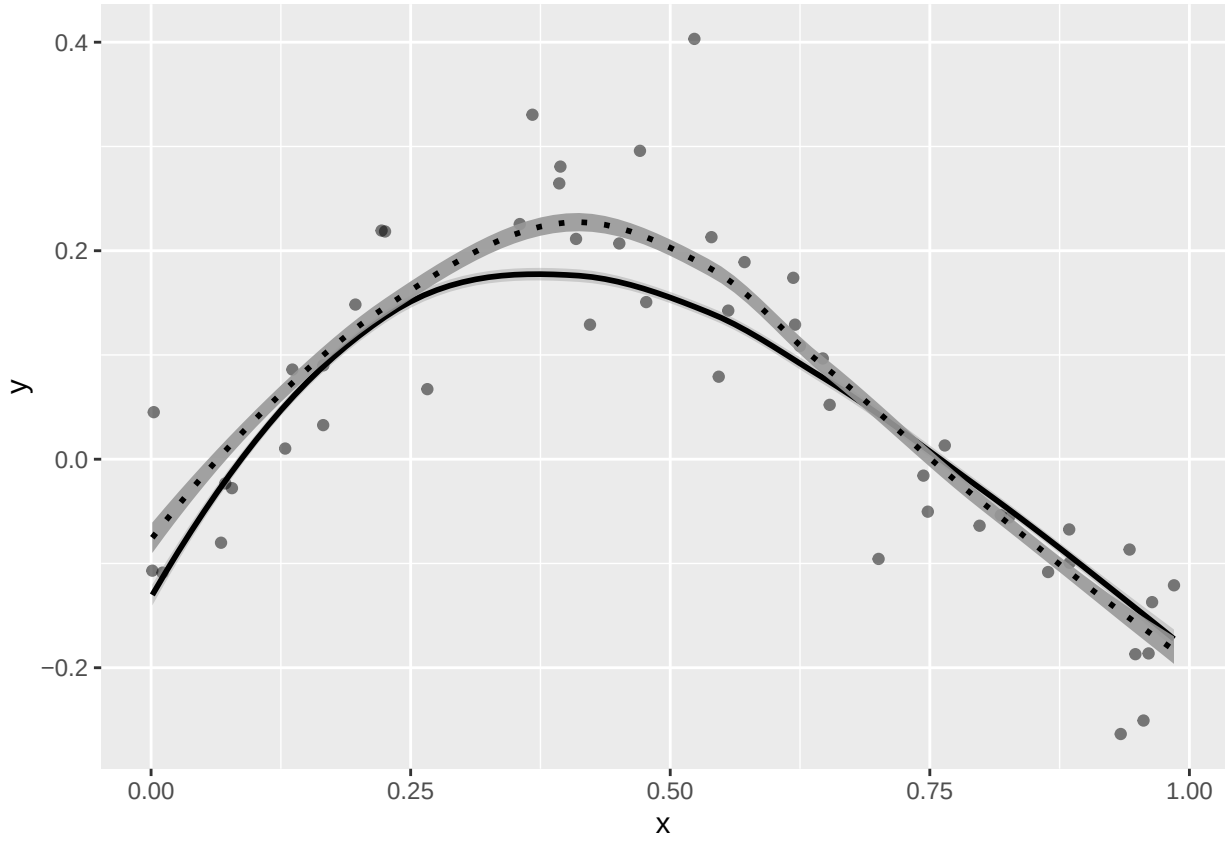
```
## Multistart 1 of 1 |Multistart 1 of 1 |Multistart 1 of 1 |Multistart 1 of 1 /Multistart 1 of 1 |
```

```
summary(model.1)
```

```
##
## Regression Data: 50 training points, in 1 variable(s)
##                   data$x
## Bandwidth(s): 0.06697485
##
## Kernel Regression Estimator: Local-Constant
## Bandwidth Type: Fixed
## Residual standard error: 0.06311353
## R-squared: 0.8386642
##
## Continuous Kernel Type: Second-Order Gaussian
## No. Continuous Explanatory Vars.: 1
```

我们对估计结果进行可视化。

```
ggplot(data=data.frame(data, fit=fitted(model.1)))+geom_point(aes(x=x,y=y),alpha=0.5)+geom_smooth(
```



2.3 Epanechnikov kernel

我们采取另外一种核函数，Epanechnikov 核函数进行估计，其具有以下形式，

$$K(u) = \frac{3}{4}(1-u^2)I(|u| \leq 1) \tag{4}$$

```
model.2 <- npreg(data$y ~ data$x,bwmethod="cv.aic",ckertype ='epanechnikov',ckerorder = 2)
```

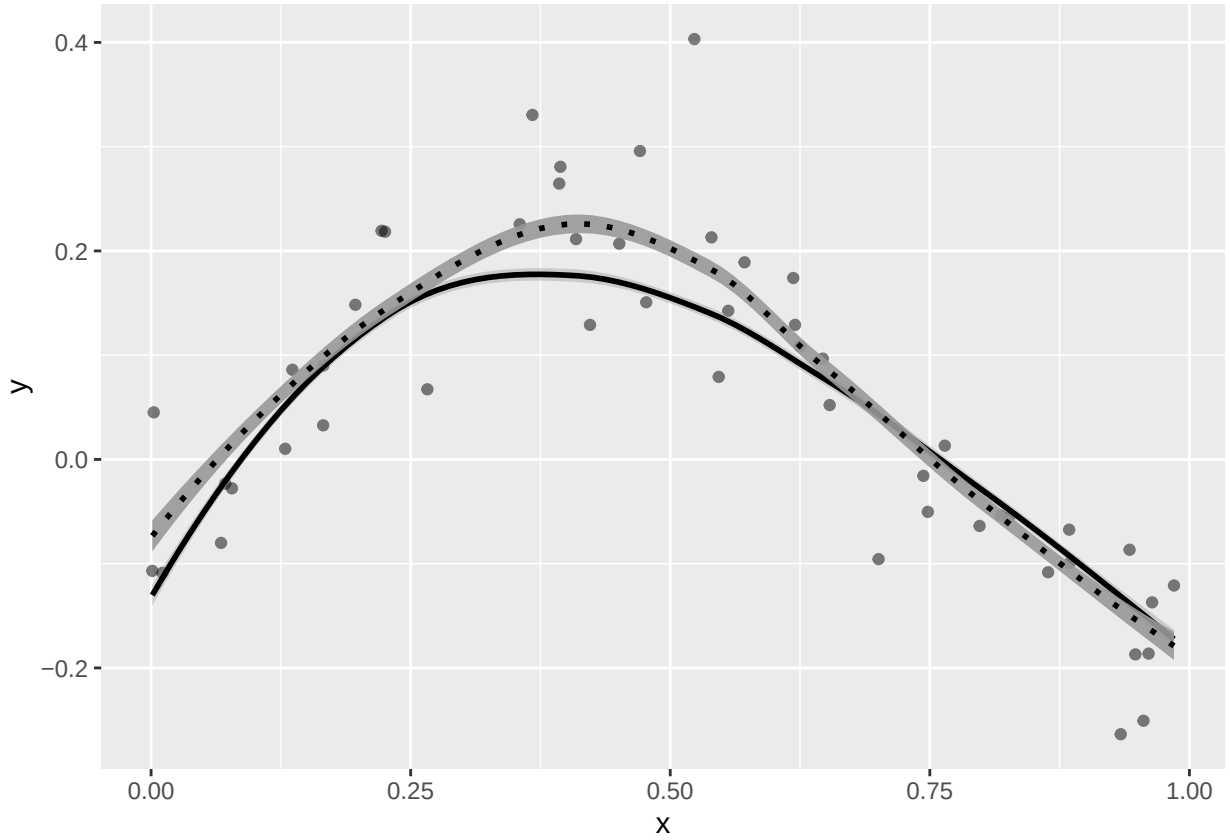
```
## Multistart 1 of 1 |Multistart 1 of 1 |Multistart 1 of 1 |Multistart 1 of 1 /Multistart 1 of 1 |
```

```
summary(model.2)
```

```
##
## Regression Data: 50 training points, in 1 variable(s)
##                  data$x
```

```
## Bandwidth(s): 0.0681954
##
## Kernel Regression Estimator: Local-Constant
## Bandwidth Type: Fixed
## Residual standard error: 0.06513644
## R-squared: 0.8283245
##
## Continuous Kernel Type: Second-Order Epanechnikov
## No. Continuous Explanatory Vars.: 1
```

```
ggplot(data=data.frame(data, fit=fitted(model.2)))+geom_point(aes(x=x,y=y),alpha=0.5)+geom_smooth(
```



从

图里我们可以发现两种估计得到的曲线都还算是理想，基本上捕捉到了曲线的特征。

对比两种核函数我们可以发现，

```
list("Goodness of fit of Gaussian Kernel"=model.1$R2,"Goodness of fit of Epanechnikov Kernel"=mode
```

```
## $`Goodness of fit of Gaussian Kernel`
## [1] 0.8386642
##
## $`Goodness of fit of Epanechnikov Kernel`
## [1] 0.8283245
```

其中高斯核函数的拟合效果更好。