

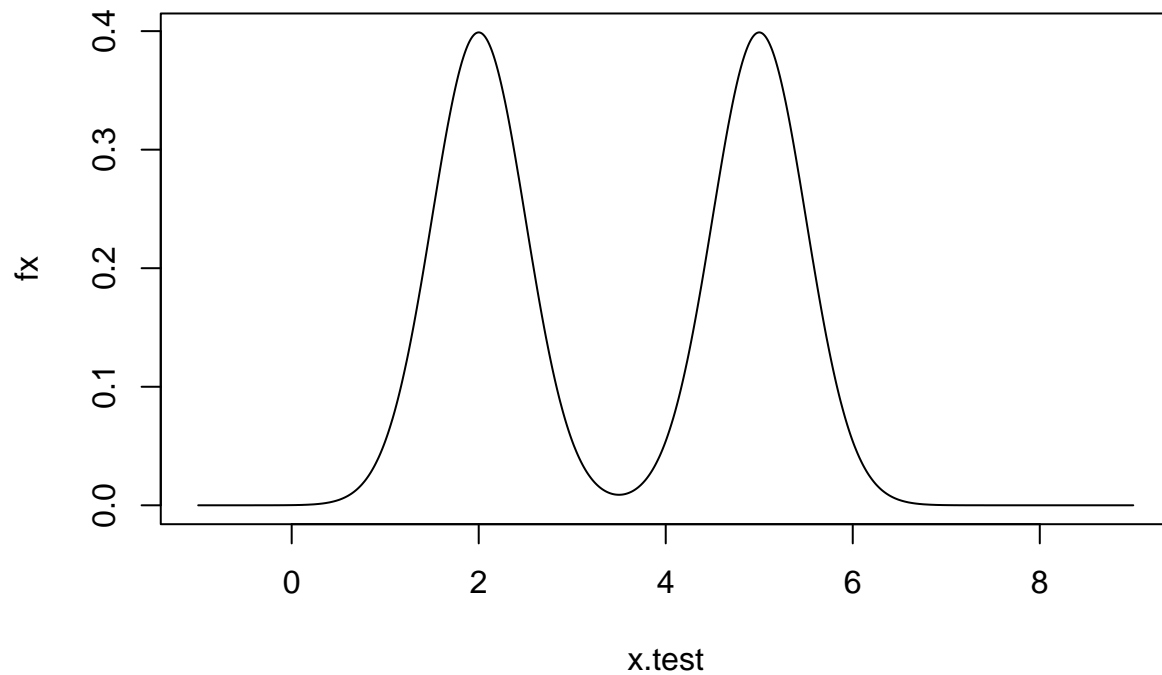
# Kernel density

Yang Wang

2/7/2023

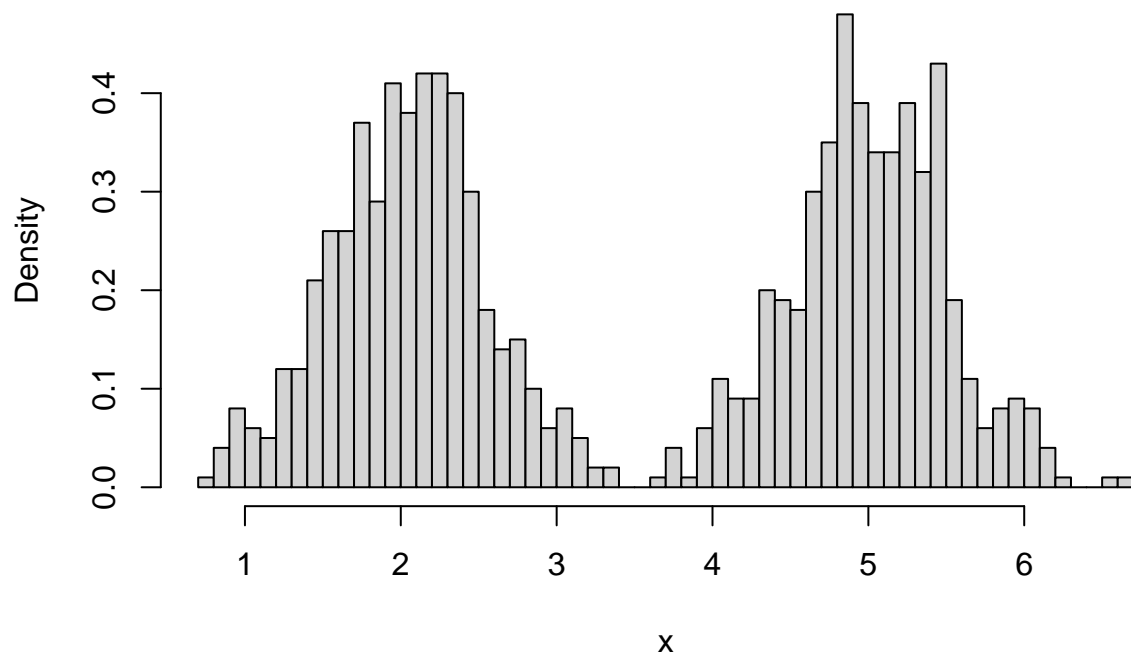
```
# Kernel density estimation
set.seed(12345)
#simulate 1000 data points from the model with the density function
#  $0.5N(2,0.5) + 0.5N(5,0.5)$ 
x = rnorm(500,2,0.5)
x = c(x, rnorm(500,5,0.5))

# the true density
x.test = seq(-1,9,0.01)
fx = 0.5*dnorm(x.test,2,0.5)+0.5*dnorm(x.test,5,0.5)
plot(x.test, fx, type = "l")
```



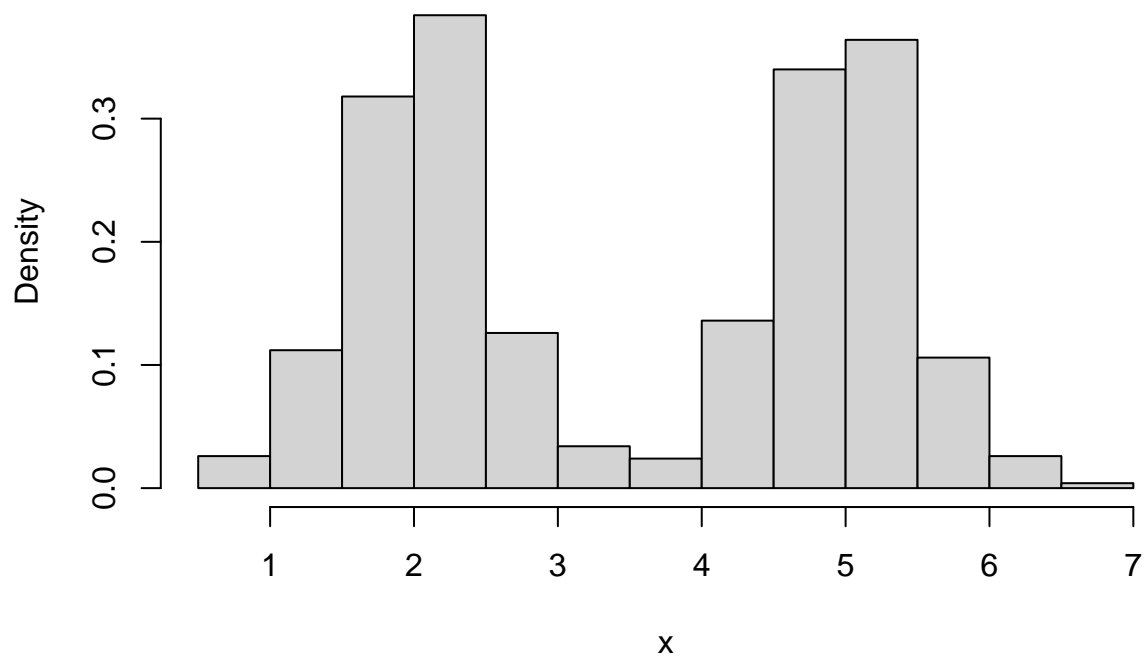
```
# histogram
hist(x,breaks = 50,freq = F)
```

**Histogram of x**



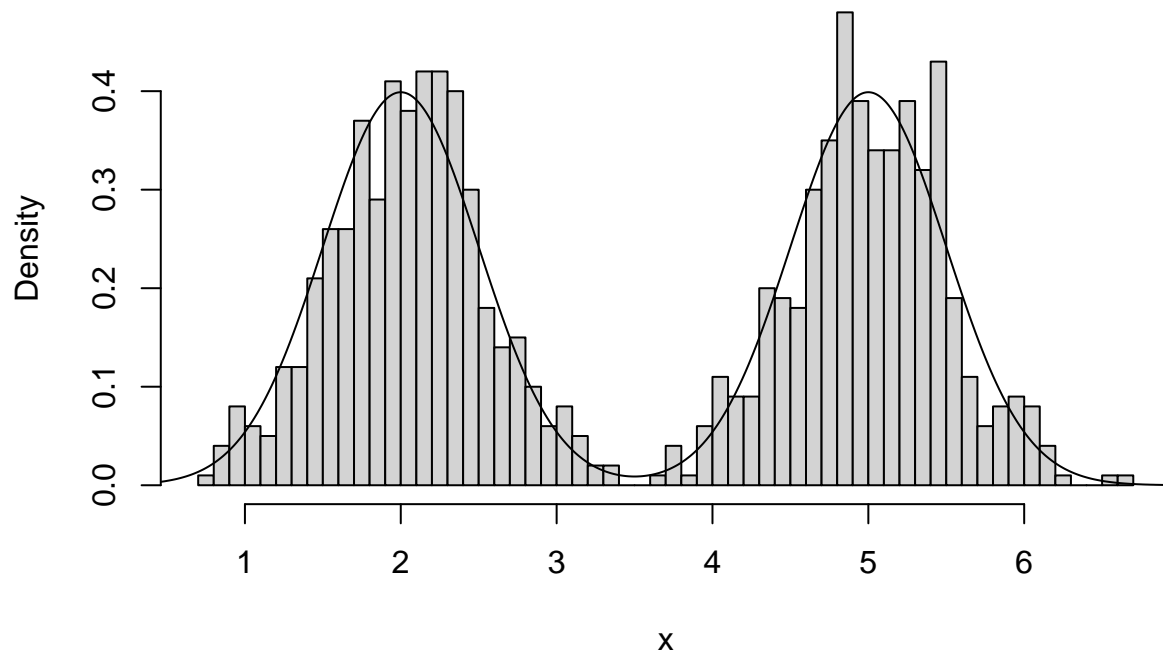
```
hist(x,breaks = 10,freq = F)
```

**Histogram of x**



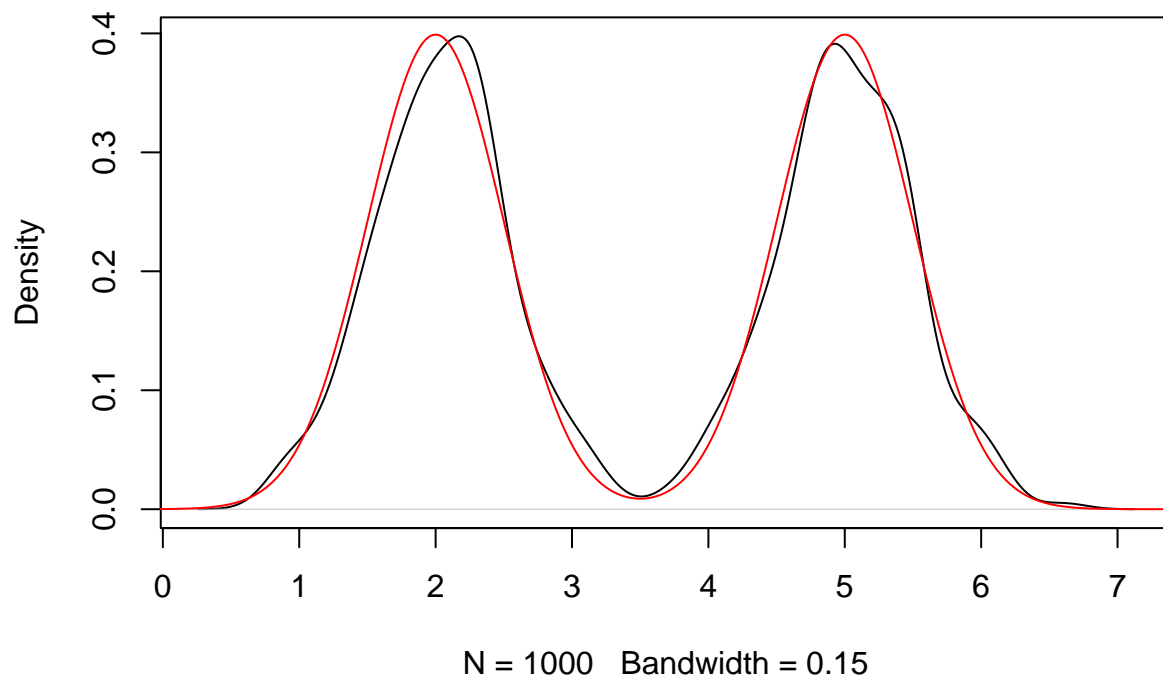
```
hist(x,breaks = 50,freq = F)  
points(x.test, fx, type = "l")
```

## Histogram of x



```
#kernel density estimate  
#bw : bandwidth  
plot(density(x, bw=0.15))  
points(x.test, fx, type = "l", col = "red")
```

**density.default(x = x, bw = 0.15)**



```

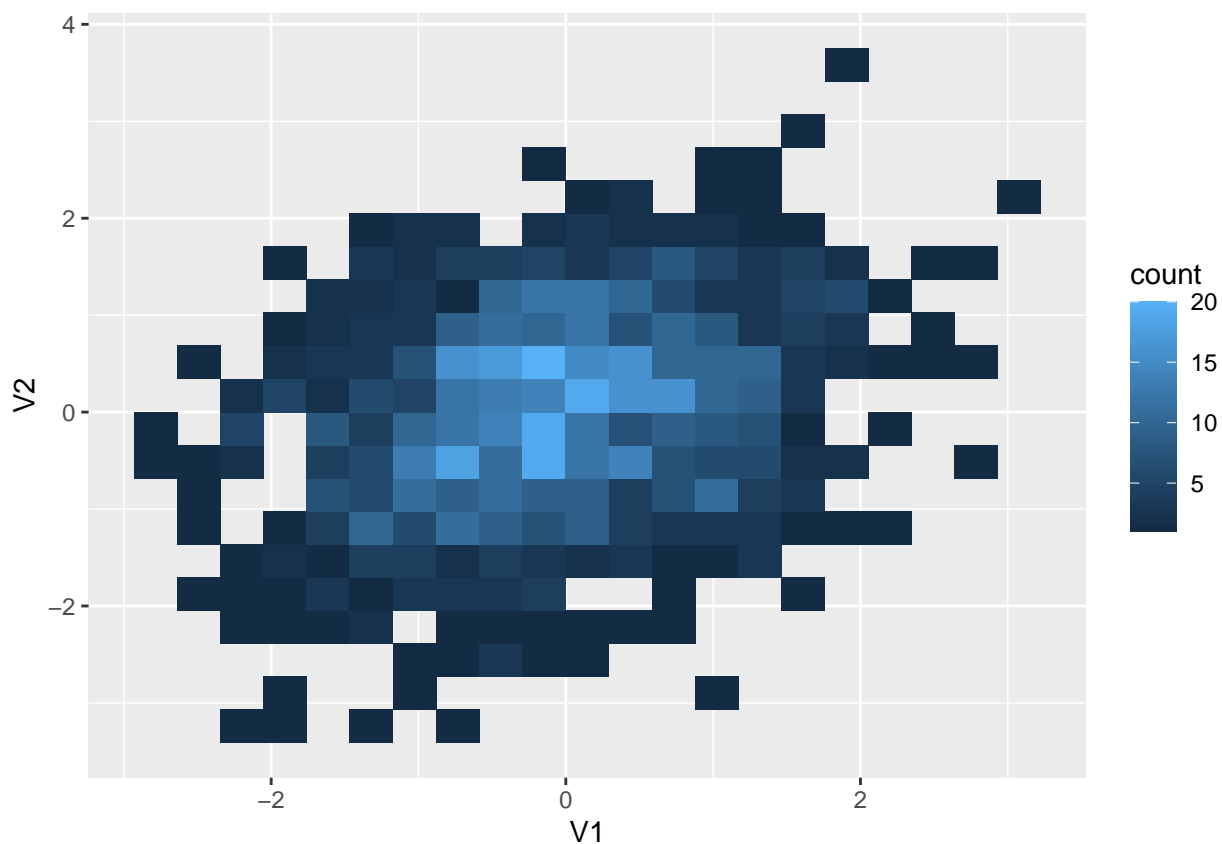
#multivariate nonparametric density estimation
library(ggplot2) # PLOT
library(mvtnorm) # Simulation
library(KernSmooth) # kernel

## KernSmooth 2.23 loaded
## Copyright M. P. Wand 1997-2009

#generate multivariate normal data
sigma = matrix(data=c(1,0.3,0.3,1), nrow = 2, ncol = 2)
x = as.data.frame(rmvnorm(1000, mean= c(0,0), sigma = sigma))

# two dimensional histogram
p = ggplot(x, aes(V1,V2))
p = p+stat_bin_2d(bins = 20)
p

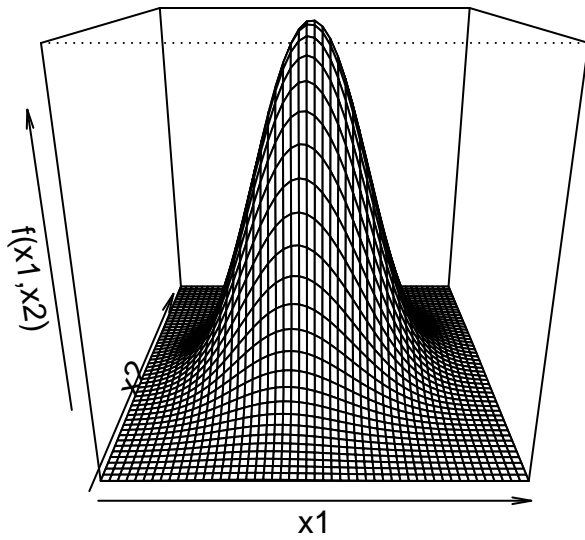
```



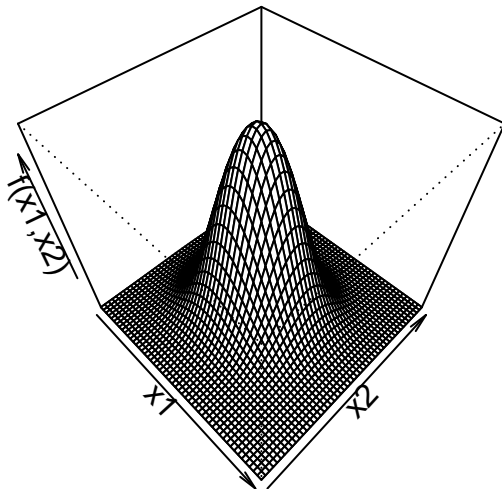
```

# two dimensional kernel density estimation
z = bkde2D(x, .9)
# three dimension plot
persp(z$fhat, theta = 0, phi = 20, xlab = "x1", ylab = "x2", zlab = "f(x1,x2)")

```



```
persp(z$fhat, theta = 45, phi = 50, xlab = "x1", ylab = "x2", zlab = "f(x1,x2)")
```



```
#contour plot of a 2d kernel density estimator  
ggplot(x, aes(V1,V2))+geom_density2d()+labs(x = "x1", y = "x2")
```

