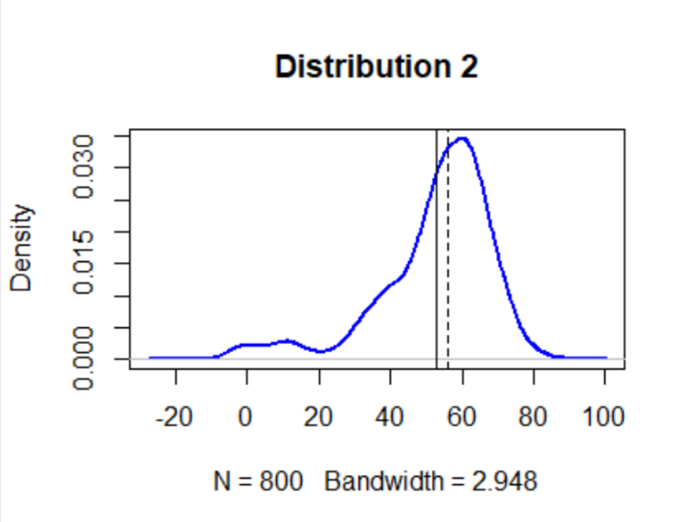
108070023 HW2

Q1

(a)

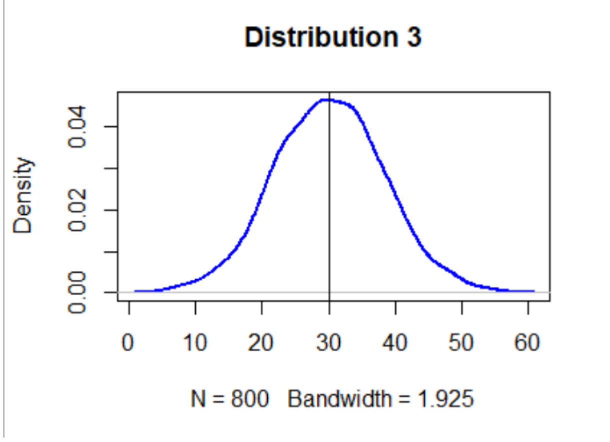
```
> # Three normally distributed data sets
> d1 <- rnorm(n=600, mean=60, sd=8)
> d2 <- rnorm(n=150, mean=40, sd=8)
> d3 <- rnorm(n=50, mean=10, sd=8)
> D2 <- c(d1, d2, d3)
> plot(density(D2), col="blue", lwd=2, main = "Distribution 2")
> abline(v=mean(D2))
> abline(v=median(D2), lty="dashed")
```



(b)

```
> D3<-rnorm(800,30,8)
> plot(density(D3), col="blue", lwd=2, main = "Distribution 3")
> abline(v=mean(D3))
> abline(v=median(D3), lty="dashed")
```

>



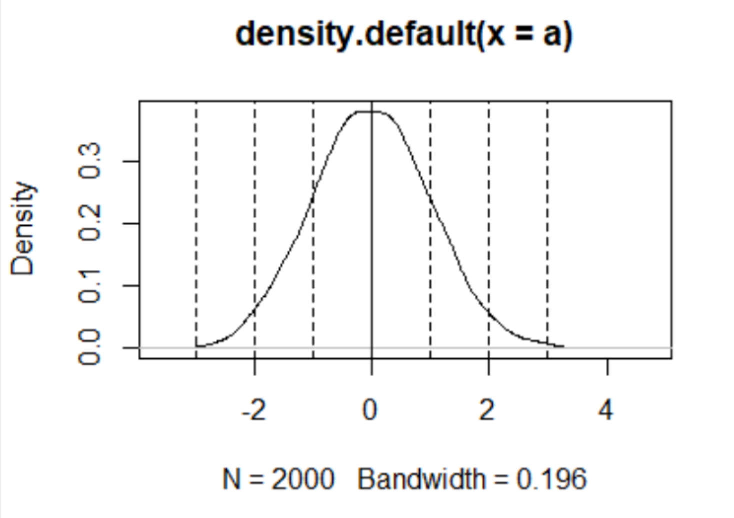
(c)

Mean is more sensitive than median,since it take the outliers into account.

Q2

(a)

```
> a<-rnorm(2000,mean=0,sd=1)
> fig1<-density(a)
> plot(fig1)
> abline(v=mean(a))
> abline(v=mean(a)+sd(a), lty="dashed")
> abline(v=mean(a)+2*sd(a), lty="dashed")
> abline(v=mean(a)+3*sd(a), lty="dashed")
> abline(v=mean(a)-sd(a), lty="dashed")
> abline(v=mean(a)-2*sd(a), lty="dashed")
> abline(v=mean(a)-3*sd(a), lty="dashed")
```



(b)

```
> st1<-quantile(a,1/4)
> st2<-quantile(a,1/2)
> st3<-quantile(a,3/4)
> sd<-sd(a)
> ans<-c(st1,st2,st3)/sd
> ans
        25%         50%         75% 
-0.69212714 -0.02438176  0.66336760 
```

(c)

```
> c<-rnorm(2000,mean=35,sd=3.5)
> st1c<-quantile(a,1/4)
> st3c<-quantile(a,3/4)
> sdc<-sd(c)
> ansc<-c(st1c,st3c)/sdc
> ans
        25%         50%         75% 
-0.69212714 -0.02438176  0.66336760 
> ansc
       25%        75% 
-0.1969997  0.1888139 
```

Standard deviations away from the mean of (c) is smaller than (b)

(d)

```
>st1d<-quantile(d123,1/4)
>st3d<-quantile(d123,3/4)
>sdd<-sd(d123)
> ans
        25%         50%         75% 
-0.69212714 -0.02438176  0.66336760 
>c(st1,st3)/sdd
        25%         75% 
-0.05831244  0.05588942 
```

Standard deviations away from the mean of (d) is smaller than (b)

Q3

(a)

Freedman-Diaconis rule is very robust and works well in practice.

(b)

```
>rand_data <- rnorm(800, mean=20, sd = 5)
> #(b)-1
> n1<-ceiling(log(800,2)+1) #num of bins
> h1<-(max(rand_data) - min(rand_data)) / n
> #(b)-2
> h2<-3.49*sd(rand_data)/(800^(1/3)) #width of bins
> n2<-ceiling((max(rand_data) - min(rand_data))/h2)
> #(b)-3
> IQR<-IQR(rand_data)
> h3=2*IQR*(800^(-1/3))#width of bins
> n3<-ceiling((max(rand_data) - min(rand_data))/h3)
> c(n1,h1)
[1] 11.00000   1.95738
> c(n2,h2)
[1] 18.000000   1.900545
> c(n3,h3)
[1] 25.000000   1.357844
```

(c)

```
> out_data <- c(rand_data, runif(10, min=40, max=60))
> #(c)-1
> out_n1<-ceiling(log(800,2)+1) #number of bins
> out_h1<-(max(out_data) - min(out_data)) / n #width of bin
> #(c)-2
> out_h2<-3.49*sd(out_data)/(800^(1/3)) #width of bins
> out_n2<-ceiling((max(out_data) - min(out_data))/h) #number of bins
> #(c)-3
> IQR<-IQR(out_data)
> out_h3=2*IQR*(800^(-1/3))#width of bins
> out_n3<-ceiling((max(out_data) - min(out_data))/h) #number of bins
> c(out_n1,out_h1)
[1] 11.000000   3.380175
> c(out_n2,out_h2)
[1] 31.00000   2.27167
> c(out_n3,out_h3)
[1] 31.000000   1.373485
> diff<-c(out_h1-h1,out_h2-h2,out_h3-h3) #calculate the difference between
(b) and (c)
```

```
> diff
[1] 1.42279476 0.37112488 0.01564072
```

Freedman-Diaconis' choice changes the least when outliers are added since it uses IQR to decide the width of bins. Therefore, the number of bins and width of bins won't be affected by outliers significantly.