STAT 443: Lab 11

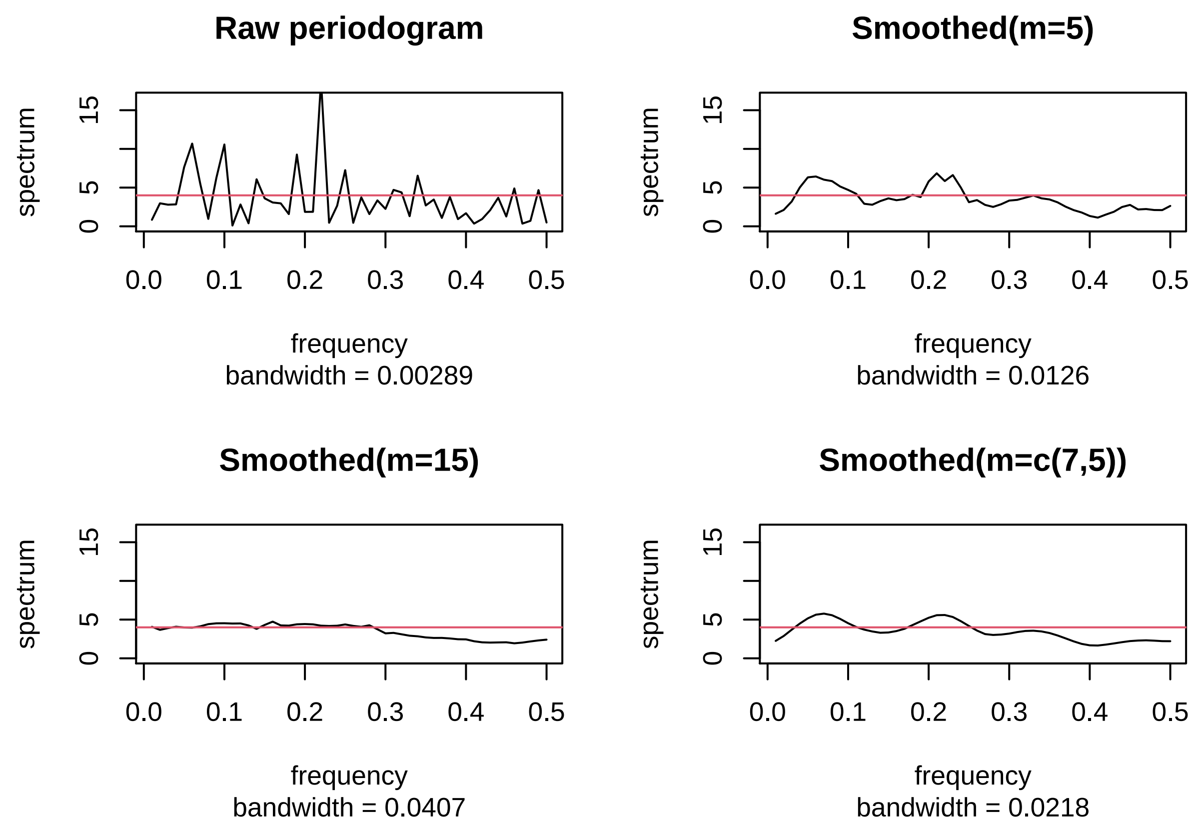
Flora Zhang 52135365

4 Apr, 2022

1.

a)

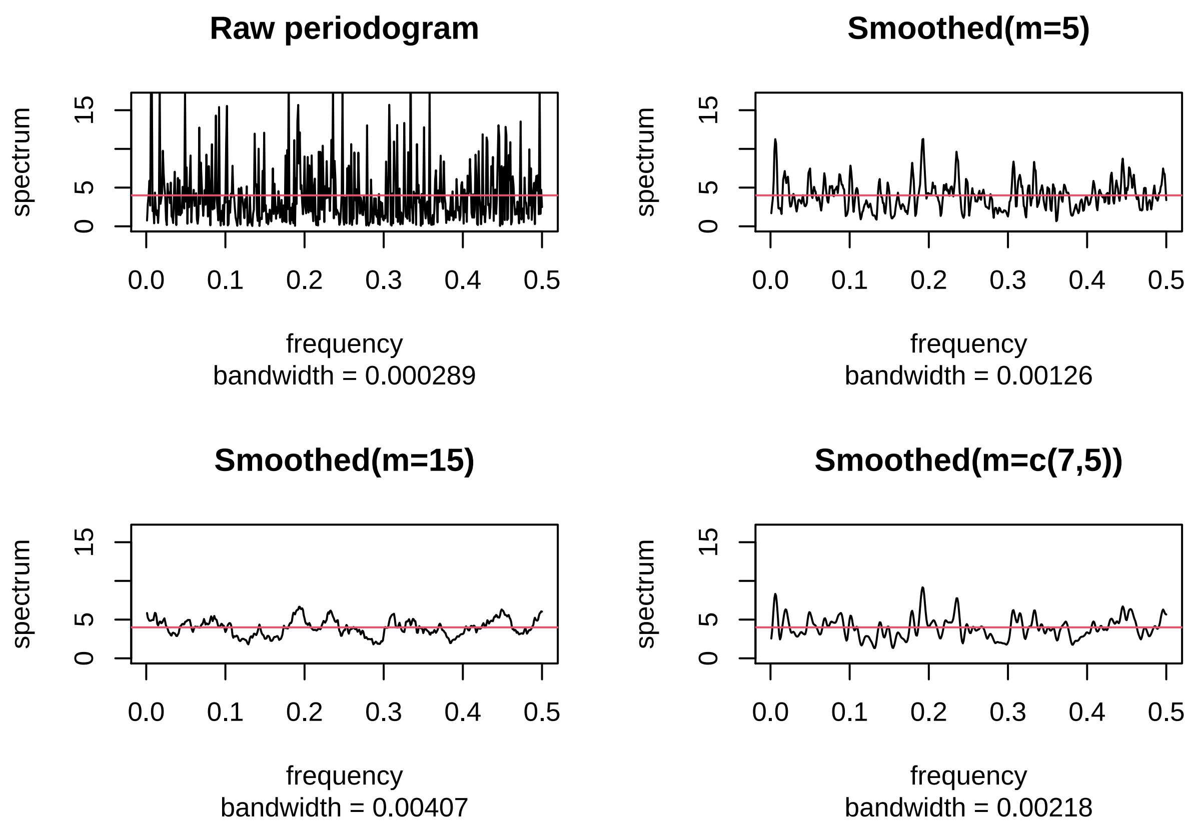
```r
set.seed(2022)
wn = arima.sim(model=list(), n=100, sd=2)

par(mfrow=c(2,2))
spec.pgram(wn, log="no", ylim=c(0,16.6), main="Raw periodogram")
abline(h=4, col=2)

#(i)
spec.pgram(wn, log="no", spans=c(5), ylim=c(0,16.6), main="Smoothed(m=5)")
abline(h=4, col=2)

#(ii)
spec.pgram(wn, log="no", spans=c(15), ylim=c(0,16.6), main="Smoothed(m=15)")
abline(h=4, col=2)

#(iii)
spec.pgram(wn, log="no", spans=c(7,5), ylim=c(0,16.6), main="Smoothed(m=c(7,5))")
abline(h=4, col=2)
```



From these plots, we see that the raw periodogram has many peaks and appears very jagged due to noise. However, it follows the overall shape of the spectral density function. In comparison, the graphs for Smoothed(m=5) and Smoothed(m=c(7,5)) appear to have less peaks and match better. The graph for Smoothed(m=15) matches the best to the spectral density function and demonstrates the least amount of peaks or fluctuations.

b)

```
set.seed(2022)
wn_1000 = arima.sim(model=list(), n=1000, sd=2)

par(mfrow=c(2,2))
spec.pgram(wn_1000, log="no", ylim=c(0,16.6), main="Raw periodogram")
abline(h=4, col=2)

#(i)
spec.pgram(wn_1000, log="no", spans=c(5), ylim=c(0,16.6), main="Smoothed(m=5)")
abline(h=4, col=2)

#(ii)
spec.pgram(wn_1000, log="no", spans=c(15), ylim=c(0,16.6), main="Smoothed(m=15)")
abline(h=4, col=2)

#(iii)
spec.pgram(wn_1000, log="no", spans=c(7,5), ylim=c(0,16.6), main="Smoothed(m=c(7,5))")
abline(h=4, col=2)
```

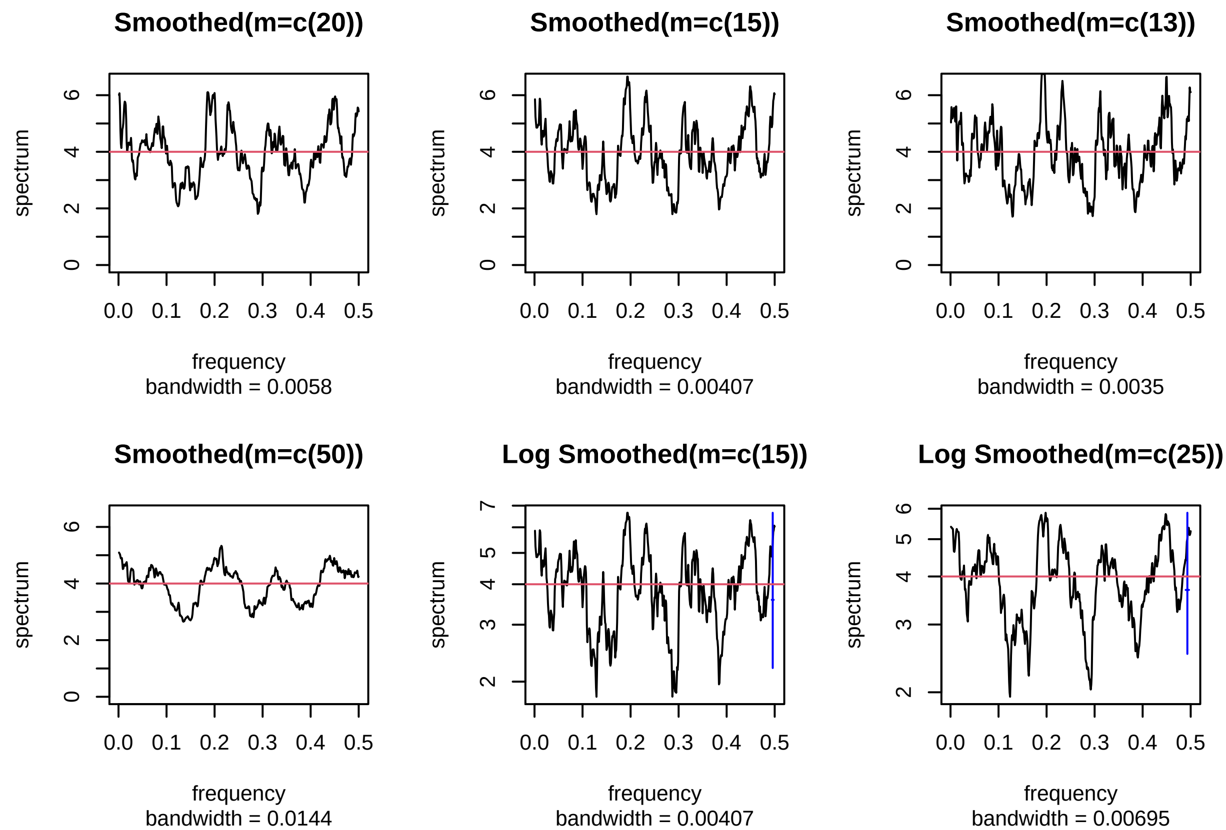


From these plots, we see that while the raw periodogram follows the overall shape of the spectral density function, it has many peaks and appears very jagged due to noise. In comparison, the graphs for Smoothed(m=5) and Smoothed(m=c(7,5)) appear to have less peaks and match better. Again, the graph for Smoothed(m=15) matches the best to the spectral density function and demonstrates the least amount of peaks or fluctuations.

c)

```
#(i)Play around with various choices of taking logs or not

par(mfrow=c(2,3))
#(ii) the spans argument, looking at no less than four special cases
spec.pgram(wn_1000, log="no", spans=c(20), ylim=c(0,6.5), main="Smoothed(m=c(20))")
abline(h=4, col=2)


spec.pgram(wn_1000, log="no", spans=c(15), ylim=c(0,6.5), main="Smoothed(m=c(15))")
abline(h=4, col=2)


spec.pgram(wn_1000, log="no", spans=c(13), ylim=c(0,6.5), main="Smoothed(m=c(13))")
abline(h=4, col=2)


spec.pgram(wn_1000, log="no", spans=c(50), ylim=c(0,6.5), main="Smoothed(m=c(50))")
abline(h=4, col=2)
```

```r
spec.pgram(wn_1000, log="yes", spans=c(15), main="Log Smoothed(m=c(15))")
abline(h=4, col=2)

spec.pgram(wn_1000, log="yes", spans=c(25), main="Log Smoothed(m=c(25))")
abline(h=4, col=2)
```



We look at cases where we take log, and those where we don't. Comparing when we take logs of the periodogram vs not, it does not appear to make a difference in terms of smoothing out the fluctuations and peaks. It appears that the plots become more smoothed as x, in spans c(x), becomes larger. For example, comparing the plot for "Smoothed(m=c(20))" and "Smoothed(m=c(50))", we see that the plot for "Smoothed(m=c(50))" is smoother and has less noise. From the cases tested here, we see that the plot, "Smoothed(m=c(50))," has the "best" fit.

d)

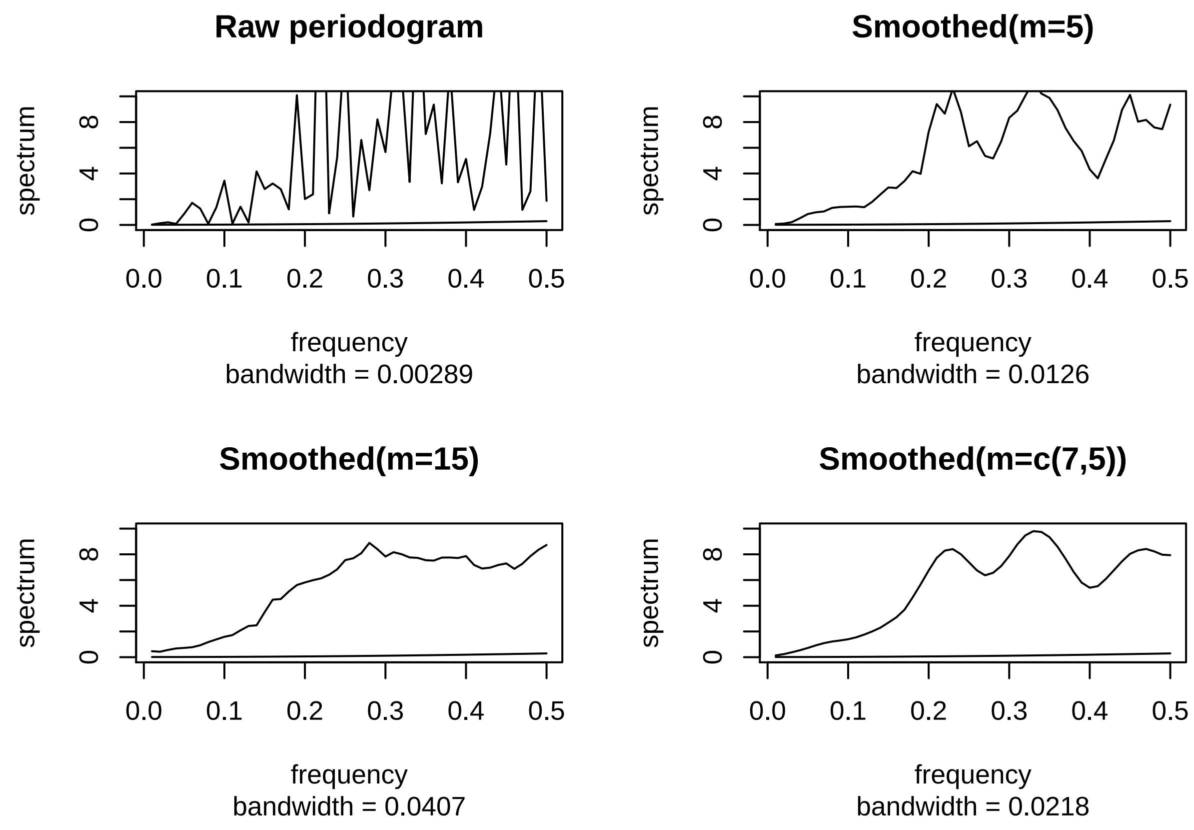
If we take logs of the relationship, we will have: $log(2 * I(w)) - log(f(w)) \sim v = log(I(w)) - log(f(w)) \sim v - c$, where v is a random variable that is related to chi-squared and c is a constant, equal to log(2).

From this equation, we can see that the error turns from multiplicative to additive, making the results easier to interpret. This is consistent with our results from (c).

2.

a)

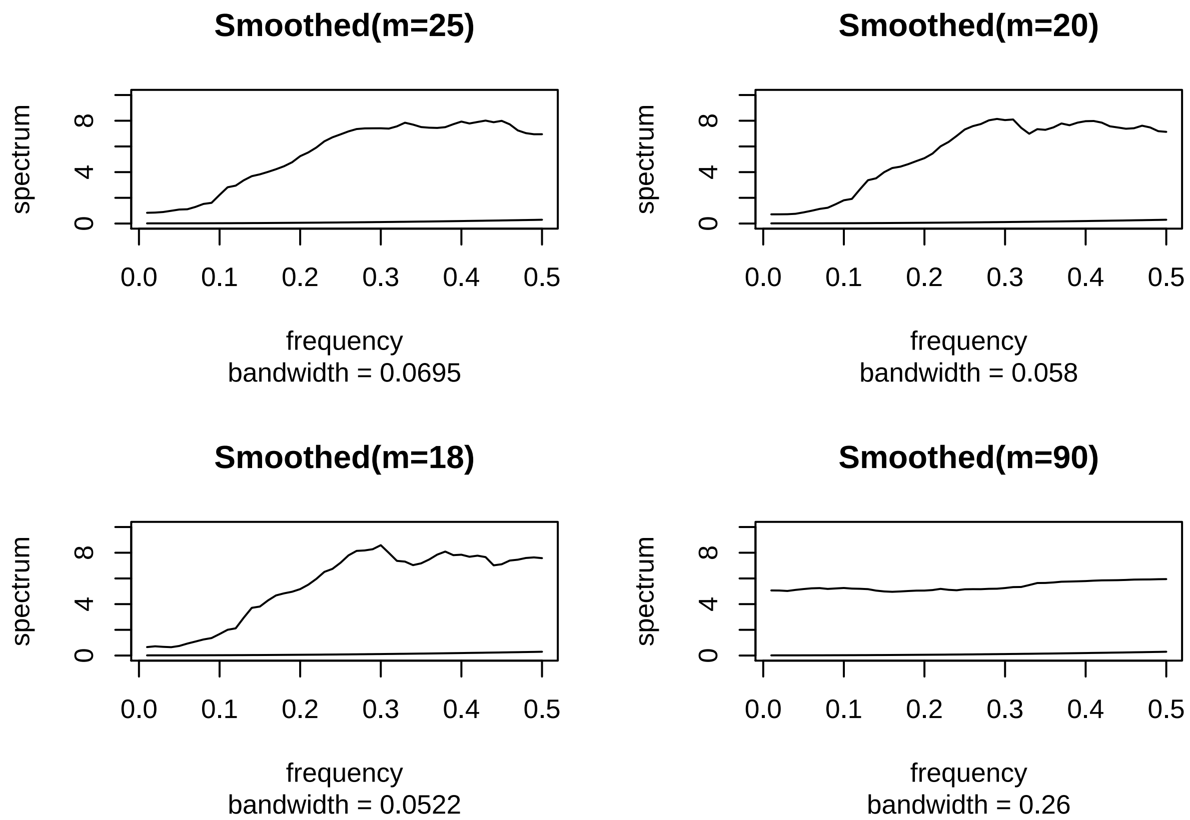
```
set.seed(2022)
wn = arima.sim(model=list(ma=c(-0.9)), n=100, sd=2)

spectral_density_2 = function(w) ((7.24/pi)*(1 - (1.8*cos(w)/1.81)))

par(mfrow=c(2,2))
spec.pgram(wn, log="no", ylim=c(0,10), main="Raw periodogram"); curve(expr=spectral_density_2, add=TRUE)

#(i)
spec.pgram(wn, log="no", spans=c(5), ylim=c(0,10), main="Smoothed(m=5)"); curve(expr=spectral_density_2

#(ii)
spec.pgram(wn, log="no", spans=c(15), ylim=c(0,10), main="Smoothed(m=15)"); curve(expr=spectral_density_

#(iii)
spec.pgram(wn, log="no", spans=c(7,5), ylim=c(0,10), main="Smoothed(m=c(7,5))"); curve(expr=spectral_den
```



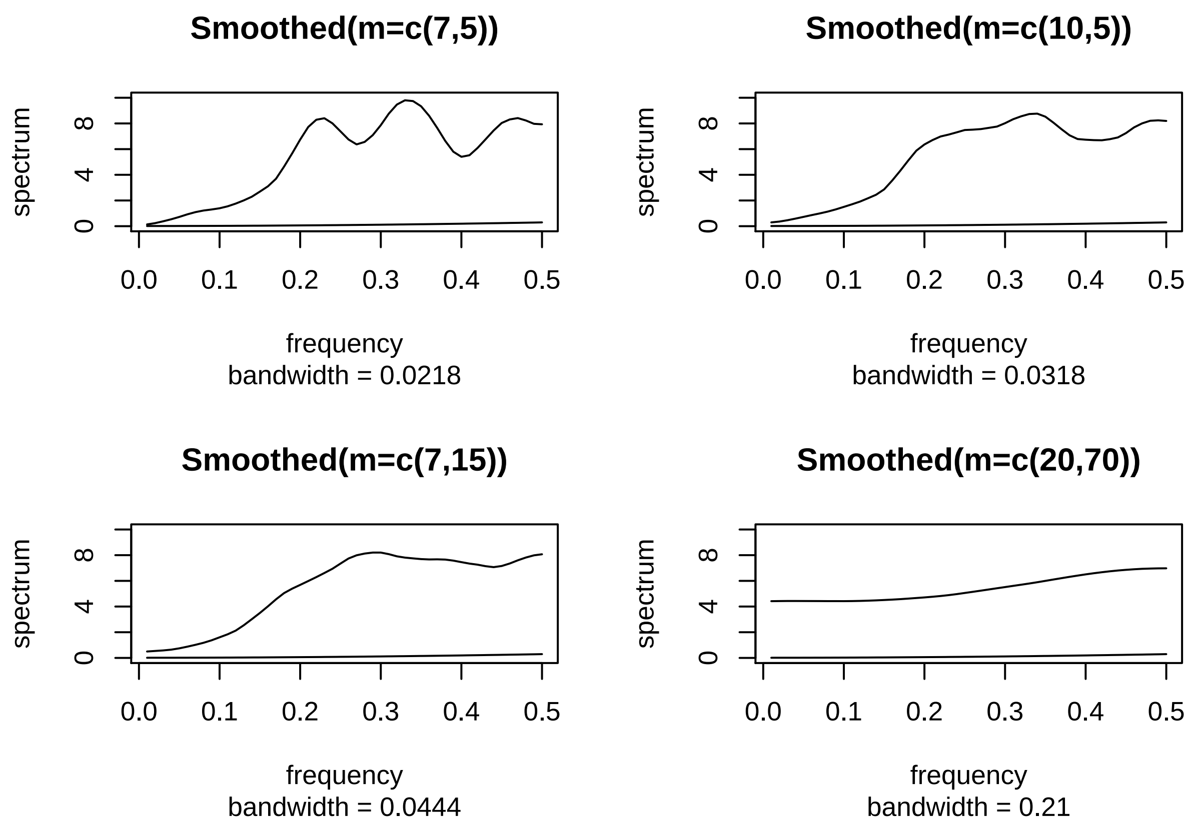
From these plots, we see that the raw periodogram has many peaks and appears very jagged due to noise. It appears to have very high fluctuations as compared to the spectral density function, and there is more noise to the right. The smoothed functions seem to have less peaks and match better, although they all still have very high fluctuations.

b) Working upwards from the choice of integers in (ii) and (iii) above, find (if you can!) both single and double smoothers that give periodograms very close to the actual spectrum

```
set.seed(2022)
wn = arima.sim(model=list(ma=c(-0.9)), n=100, sd=2)

spectral_density_2 = function(w) ((7.24/pi)*(1 - (1.8*cos(w)/1.81)))

#(ii)
par(mfrow=c(2,2))
spec.pgram(wn, log="no", spans=c(25), ylim=c(0,10), main="Smoothed(m=25)"); curve(expr=spectral_density_

spec.pgram(wn, log="no", spans=c(20), ylim=c(0,10), main="Smoothed(m=20)"); curve(expr=spectral_density_

spec.pgram(wn, log="no", spans=c(18), ylim=c(0,10), main="Smoothed(m=18)"); curve(expr=spectral_density_

spec.pgram(wn, log="no", spans=c(90), ylim=c(0,10), main="Smoothed(m=90)"); curve(expr=spectral_density_
```



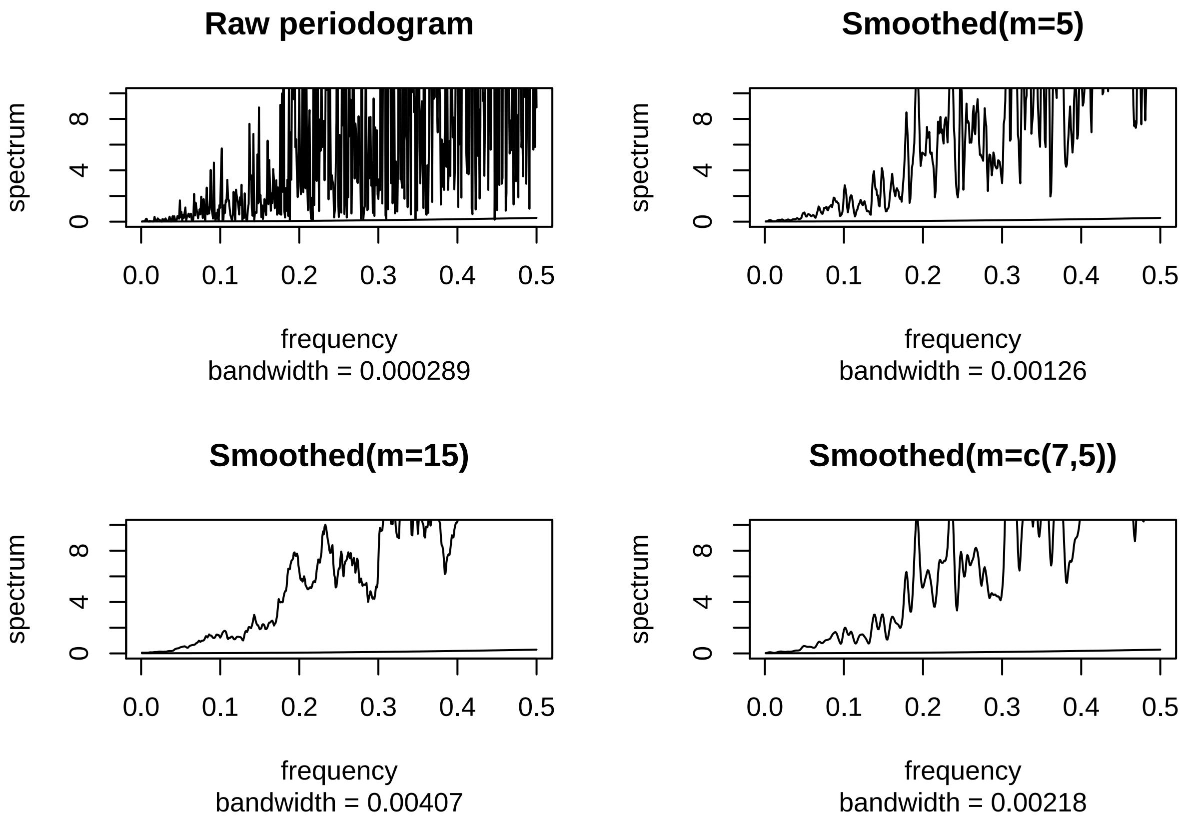

```
#(iii)
spec.pgram(wn, log="no", spans=c(7,5), ylim=c(0,10), main="Smoothed(m=c(7,5))"); curve(expr=spectral_den

spec.pgram(wn, log="no", spans=c(10,5), ylim=c(0,10), main="Smoothed(m=c(10,5))"); curve(expr=spectral_

spec.pgram(wn, log="no", spans=c(7,15), ylim=c(0,10), main="Smoothed(m=c(7,15))"); curve(expr=spectral_

spec.pgram(wn, log="no", spans=c(20,70), ylim=c(0,10), main="Smoothed(m=c(20,70))"); curve(expr=spectral
```



It appears that Smoothed(m=90) and Smoothed(m=c(20,70)) follow the overall shape the best, although there is still noise.

c)

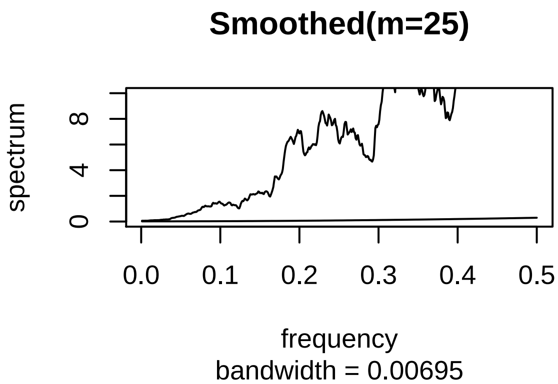
```
set.seed(2022)
wn = arima.sim(model=list(ma=c(-0.9)), n=1000, sd=2)

spectral_density_2 = function(w) ((7.24/pi)*(1 - (1.8*cos(w)/1.81)))

par(mfrow=c(2,2))
spec.pgram(wn, log="no", ylim=c(0,10), main="Raw periodogram"); curve(expr=spectral_density_2, add=TRUE

#(i)
spec.pgram(wn, log="no", spans=c(5), ylim=c(0,10), main="Smoothed(m=5)"); curve(expr=spectral_density_2

#(ii)
spec.pgram(wn, log="no", spans=c(15), ylim=c(0,10), main="Smoothed(m=15)"); curve(expr=spectral_density

#(iii)
spec.pgram(wn, log="no", spans=c(7,5), ylim=c(0,10), main="Smoothed(m=c(7,5))"); curve(expr=spectral_de
```

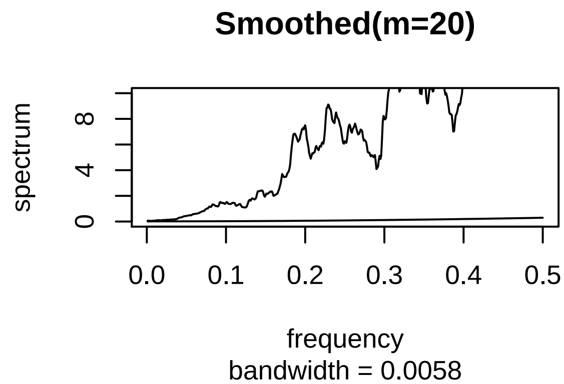


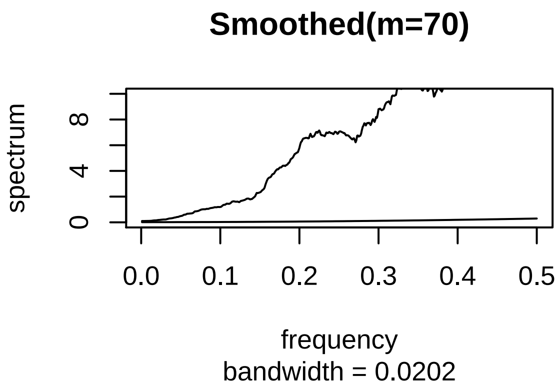
From these plots, we see that the raw periodogram has many peaks and appears very jagged due to noise. It appears to have very high fluctuations as compared to the spectral density function. The smoothed functions seem to have less peaks and match better, although they all still have very high fluctuations.

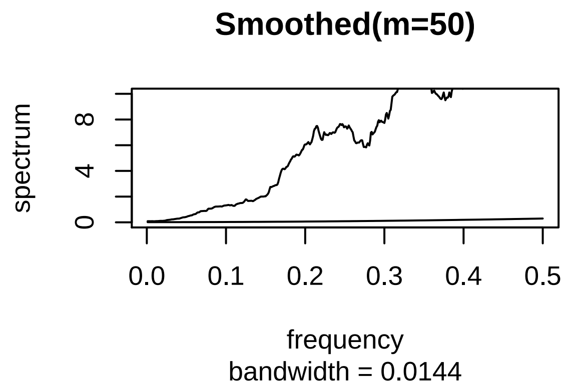
d) Working upwards from the choice of integers in (ii) and (iii) above, find (if you can!) both single and double smoothers that give periodograms very close to the actual spectrum. Comment on your results.

```
set.seed(2022)
wn = arima.sim(model=list(ma=c(-0.9)), n=1000, sd=2)

spectral_density_2 = function(w) ((7.24/pi)*(1 - (1.8*cos(w)/1.81)))

#(ii)
par(mfrow=c(2,2))
spec.pgram(wn, log="no", spans=c(25), ylim=c(0,10), main="Smoothed(m=25)"); curve(expr=spectral_density

spec.pgram(wn, log="no", spans=c(20), ylim=c(0,10), main="Smoothed(m=20)"); curve(expr=spectral_density

spec.pgram(wn, log="no", spans=c(70), ylim=c(0,10), main="Smoothed(m=70)"); curve(expr=spectral_density

spec.pgram(wn, log="no", spans=c(50), ylim=c(0,10), main="Smoothed(m=50)"); curve(expr=spectral_density
```

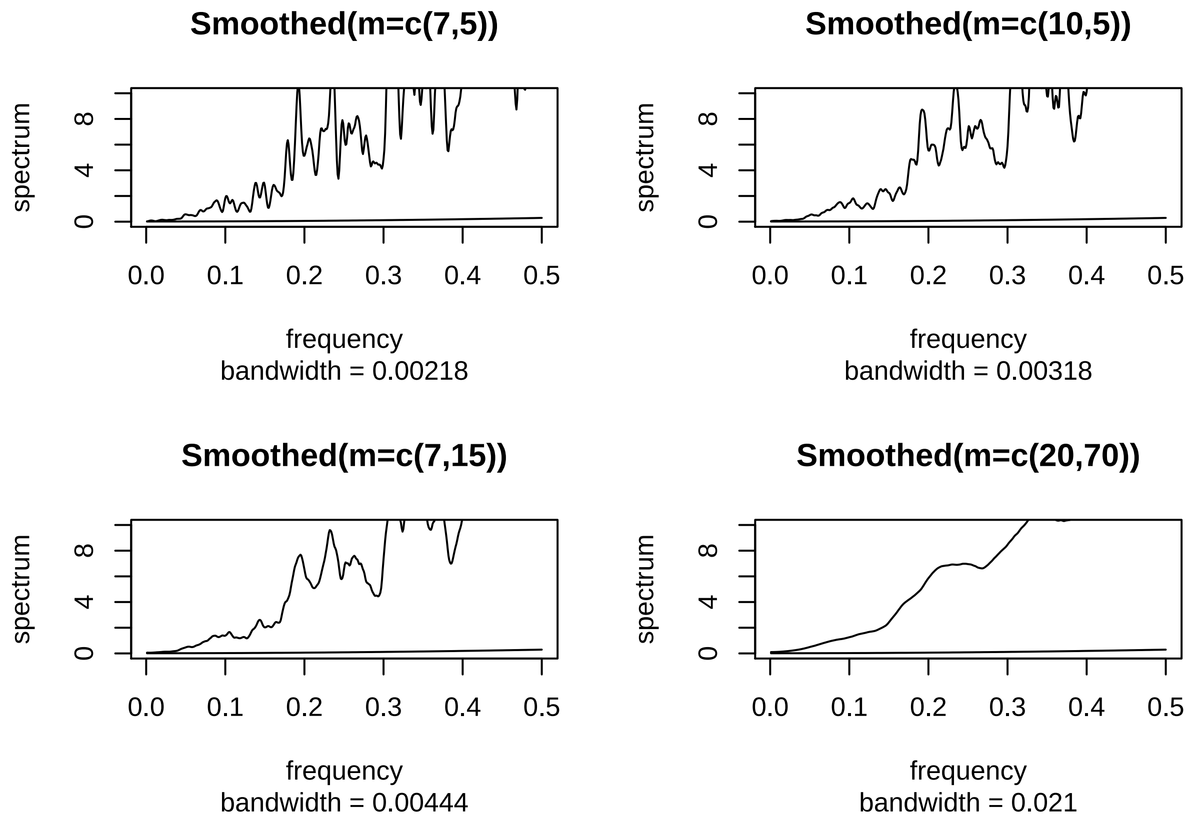









```
#(iii)
spec.pgram(wn, log="no", spans=c(7,5), ylim=c(0,10), main="Smoothed(m=c(7,5))"); curve(expr=spectral_der

spec.pgram(wn, log="no", spans=c(10,5), ylim=c(0,10), main="Smoothed(m=c(10,5))"); curve(expr=spectral_

spec.pgram(wn, log="no", spans=c(7,15), ylim=c(0,10), main="Smoothed(m=c(7,15))"); curve(expr=spectral_

spec.pgram(wn, log="no", spans=c(20,70), ylim=c(0,10), main="Smoothed(m=c(20,70))"); curve(expr=spectral
```



It appears like Smoothed(m=c(20,70)) and Smoothed(m=c(70)) fit the model the best.