Homework 9

Zahlen Zbinden

2024-03-09

```r
library(TSA)
```

```
Warning: package 'TSA' was built under R version 4.3.2


Attaching package: 'TSA'

The following objects are masked from 'package:stats':

    acf, arima

The following object is masked from 'package:utils':

    tar
```

```r
library(tidyverse)
```

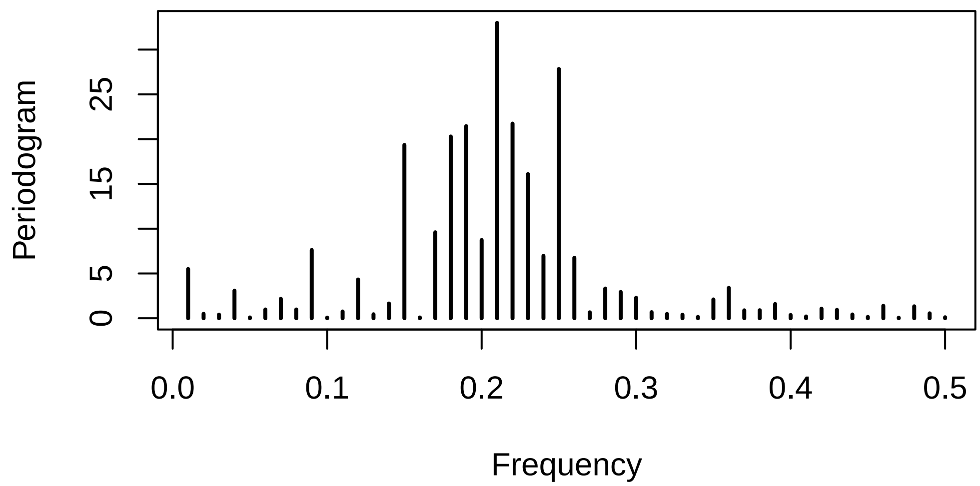
```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.3     v readr     2.1.4
v forcats   1.0.0     v stringr   1.5.0
v ggplot2   3.4.3     v tibble    3.2.1
v lubridate 1.9.2     v tidyr     1.3.0
v purrr     1.0.2

-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
x readr::spec()   masks TSA::spec()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
```

```r
library(ggplot2)
```

1) consider an AR(2) model $Y_t = .5Y_{t-1} - .8Y_{t-2} + W_t$ where W_t is iid N(0,1)

a. simulate a series with n=100, and use periodogram to obtain sample spectral density

```r
n <- 100
y1 <- arima.sim(n=n, model=list(ar=c(.5, -.8)))
per_y1 <- periodogram(y1, log="no")
```



b. repeat a 1000 times, for each replication, save the sample spectral density

```r
nsim <- 1000
spec_sim <- matrix(NA, nsim, length(per_y1$freq))
```

```r
for(i in 1:nsim) {
    y1 <- arima.sim(n=n, model=list(ar=c(.5, -.8)))
    spec_sim[i,] <- periodogram(y1, plot=FALSE)$spec
}
```

c. based on sample spectral desnsities obtained in 1000 replications, calculate the average sample density and the standard deviation for each frequency

```r
spec_avg <- apply(spec_sim, 2, mean)
spec_sd <- apply(spec_sim, 2, sd)
spec_t <- ARMAspec(model=list(ar=c(.5, -.8)), freq=per_y1$freq, plot=FALSE)
```

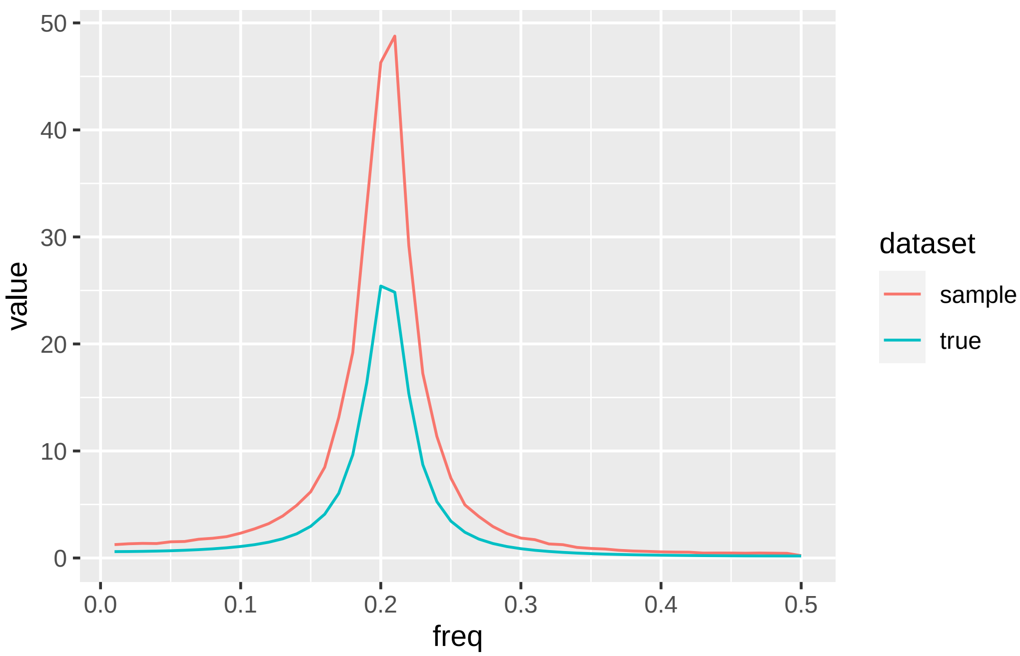
d. Plot the averages sample density by frequency and compare it with the true spectral density. Also plot the standard deviation by frequency

```r
savg_plot <- spec_avg |>
              as_tibble() |>
              mutate(freq=per_y1$freq, dataset="sample")

t_plot <- spec_t$spec |>
              as_tibble() |>
              mutate(freq=per_y1$freq, dataset="true") |>
              rename(value="V1")
```

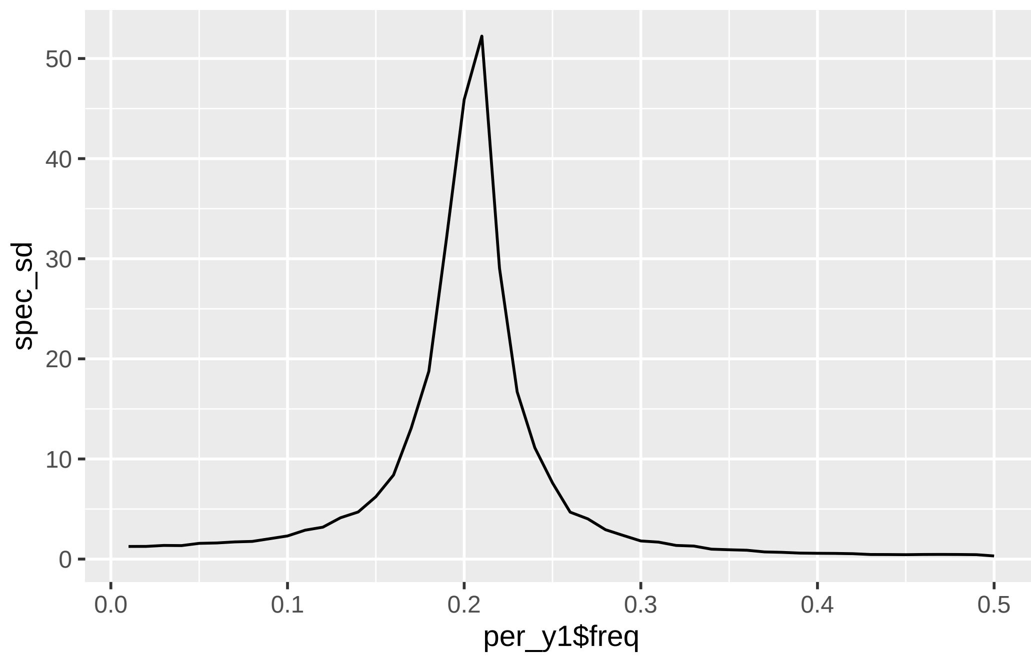
```
Warning: The `x` argument of `as_tibble.matrix()` must have unique column names if
`.name_repair` is omitted as of tibble 2.0.0.
i Using compatibility `.name_repair`.
```

```r
plot_data <- bind_rows(savg_plot, t_plot)
```

```r
plot_data |>
  ggplot(aes(x=freq, y=value, color=dataset)) +
    geom_line()
```



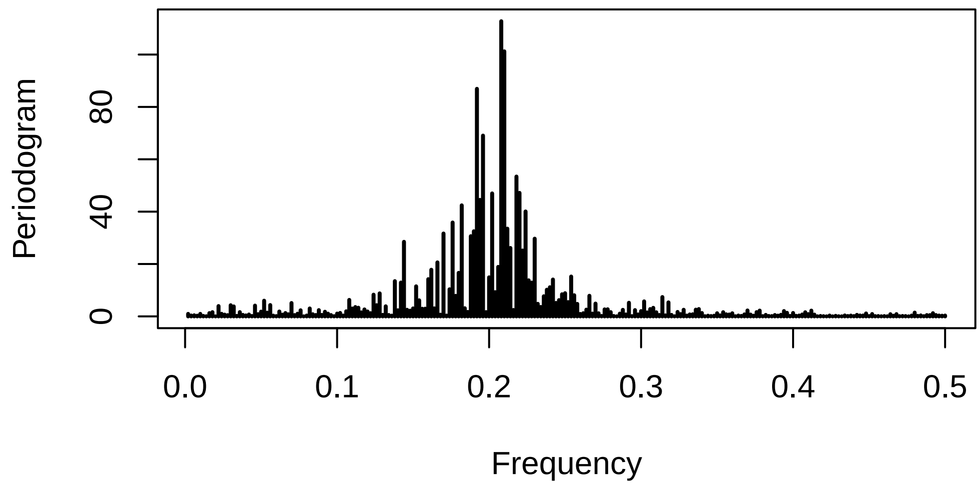

```
spec_sd |>
  as_tibble() |>
  mutate(freq=per_y1$freq) |>
  ggplot(aes(x=per_y1$freq, y=spec_sd)) +
    geom_line()
```



e. repeat a-d for sample sizes n=500 and n=1000. From the plots of average sample density and the plots of the standard deviatsion, what can you condlude about the performance of periodogram? In particular, is the periodogram consistent?

n = 500

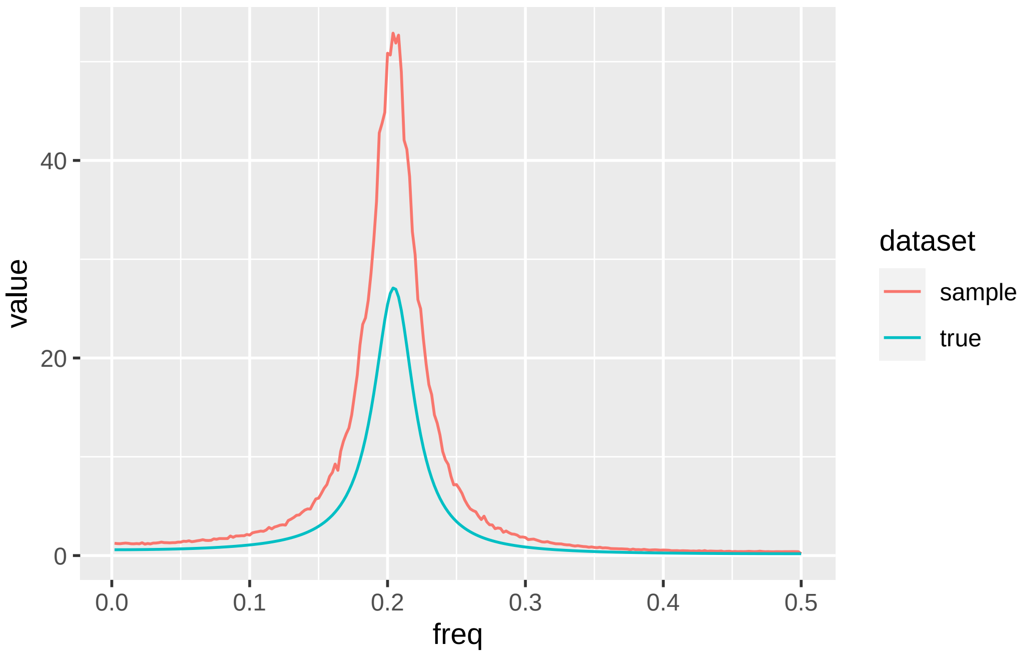
```
n <- 500
y1 <- arima.sim(n=n, model=list(ar=c(.5, -.8)))
per_y1 <- periodogram(y1, log="no")
```



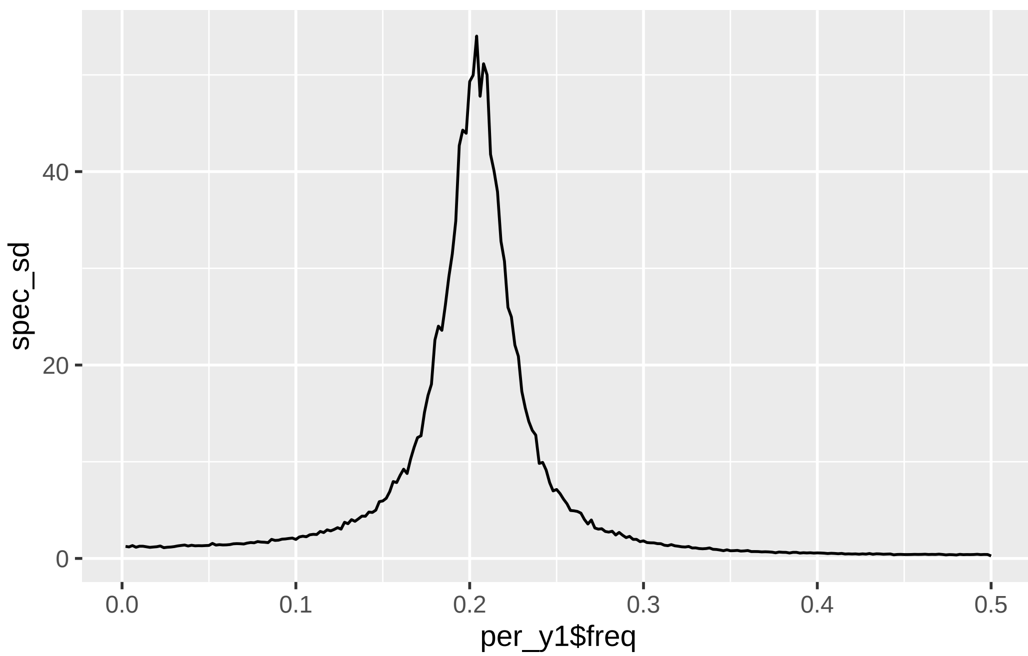

```
nsim <- 1000
spec_sim <- matrix(NA, nsim, length(per_y1$freq))
for(i in 1:nsim) {
    y1 <- arima.sim(n=n, model=list(ar=c(.5, -.8)))
    spec_sim[i,] <- periodogram(y1, plot=FALSE)$spec
}
spec_avg <- apply(spec_sim, 2, mean)
spec_sd <- apply(spec_sim, 2, sd)
spec_t <- ARMAspec(model=list(ar=c(.5, -.8)), freq=per_y1$freq, plot=FALSE)
```

```
savg_plot <- spec_avg |>
              as_tibble() |>
              mutate(freq=per_y1$freq, dataset="sample")

t_plot <- spec_t$spec |>
              as_tibble() |>
              mutate(freq=per_y1$freq, dataset="true") |>
              rename(value="V1")

plot_data <- bind_rows(savg_plot, t_plot)
```

```
plot_data |>
  ggplot(aes(x=freq, y=value, color=dataset)) +
    geom_line()
```

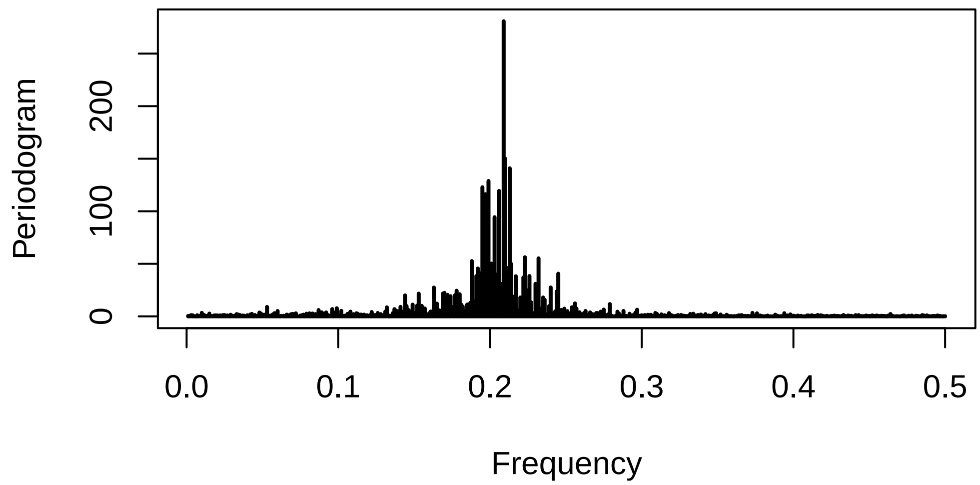



```
spec_sd |>
  as_tibble() |>
  mutate(freq=per_y1$freq) |>
  ggplot(aes(x=per_y1$freq, y=spec_sd)) +
    geom_line()
```



n = 1000

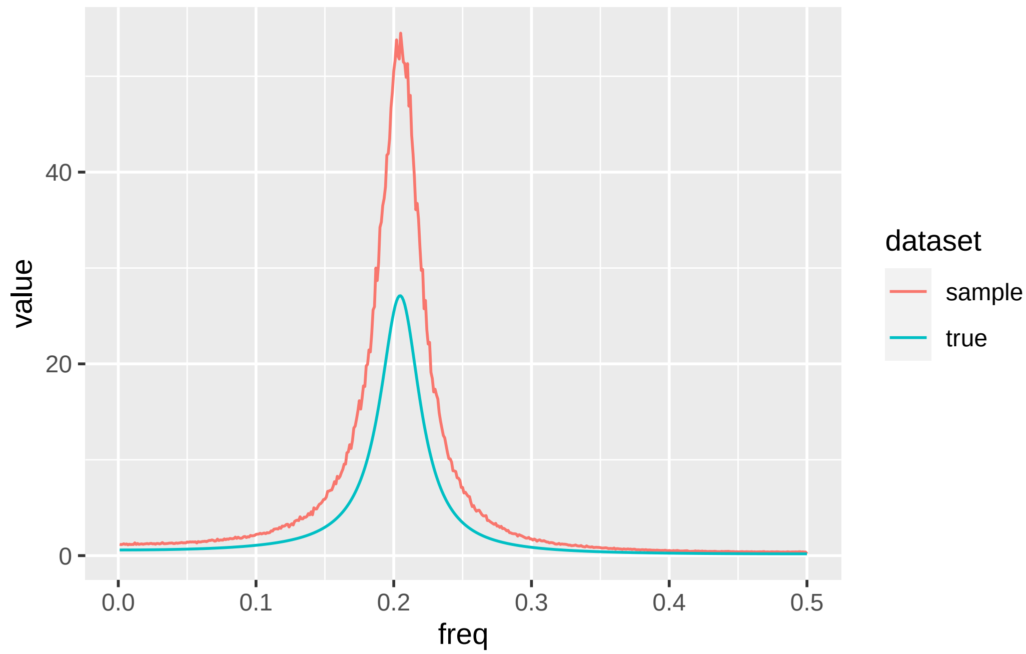
```
n <- 1000
y1 <- arima.sim(n=n, model=list(ar=c(.5, -.8)))
per_y1 <- periodogram(y1, log="no")
```



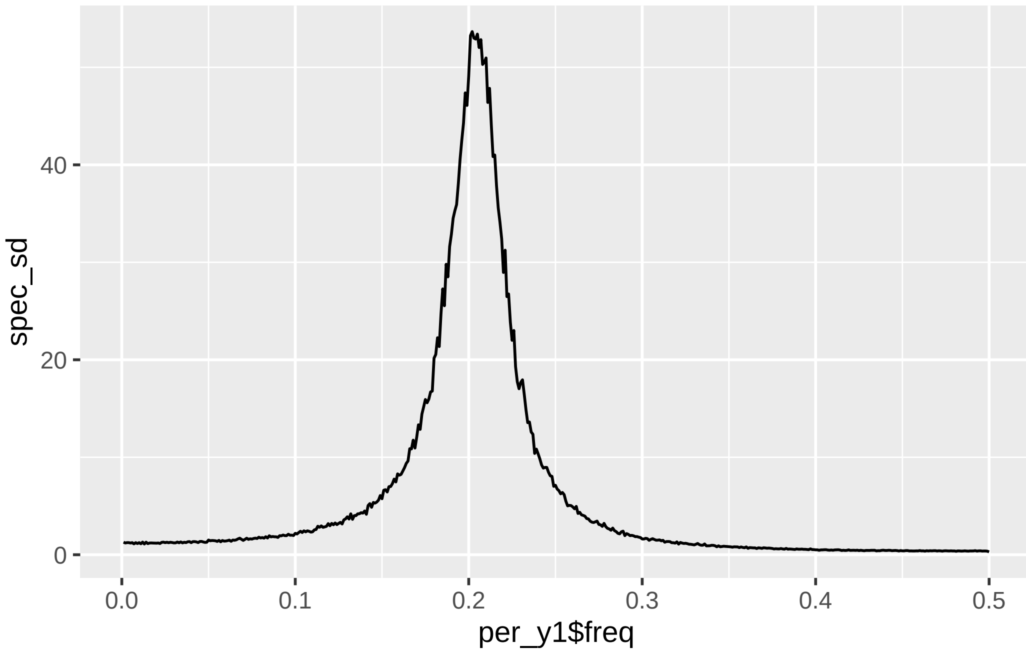

```
nsim <- 1000
spec_sim <- matrix(NA, nsim, length(per_y1$freq))
for(i in 1:nsim) {
    y1 <- arima.sim(n=n, model=list(ar=c(.5, -.8)))
    spec_sim[i,] <- periodogram(y1, plot=FALSE)$spec
}
spec_avg <- apply(spec_sim, 2, mean)
spec_sd <- apply(spec_sim, 2, sd)
spec_t <- ARMAspec(model=list(ar=c(.5, -.8)), freq=per_y1$freq, plot=FALSE)
```

```
savg_plot <- spec_avg |>
              as_tibble() |>
              mutate(freq=per_y1$freq, dataset="sample")

t_plot <- spec_t$spec |>
              as_tibble() |>
              mutate(freq=per_y1$freq, dataset="true") |>
              rename(value="V1")

plot_data <- bind_rows(savg_plot, t_plot)
```

```r
plot_data |>
  ggplot(aes(x=freq, y=value, color=dataset)) +
    geom_line()
```




```r
spec_sd |>
  as_tibble() |>
  mutate(freq=per_y1$freq) |>
  ggplot(aes(x=per_y1$freq, y=spec_sd)) +
    geom_line()
```



The three plots are very similar for differing sample sizes. Especially the standard deviation does not decrease as the sample size increases. This indicates that the periodogram is not consistent.

2) Consdier LA rainfall. Because of the skewness in series, we will use the logarithms of the raw values.

```
data(larain)
head(larain)
```

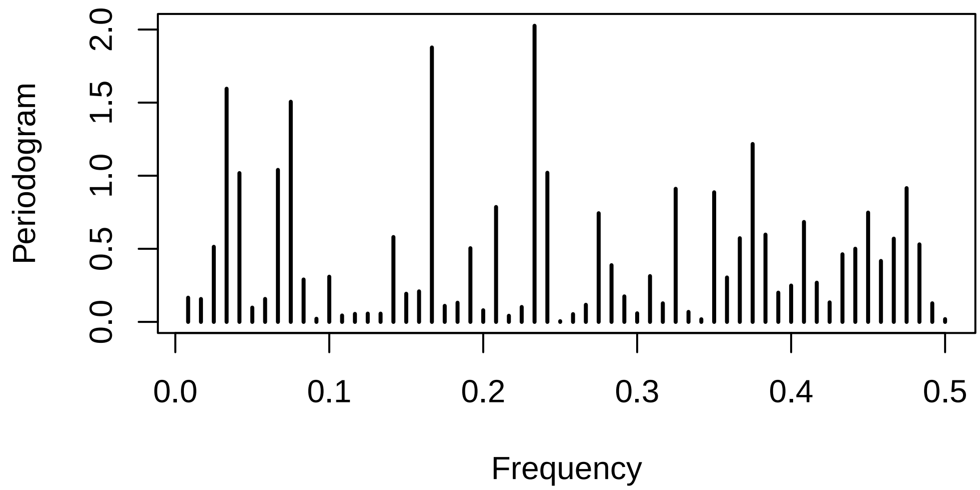
```
[1] 20.86 17.41 18.65  5.53 10.74 14.14
```

```
log_rain <- log(larain)
head(log_rain)
```

```
[1] 3.037833 2.857045 2.925846 1.710188 2.373975 2.649008
```

a. use periodogram() to produce raw periodogram for logarithms of rainfall values

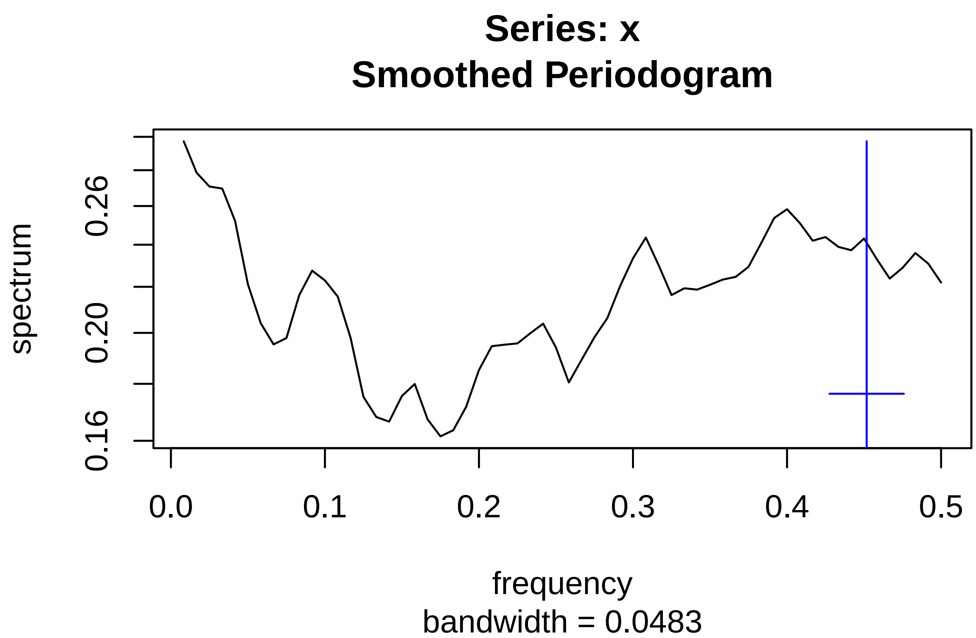
```
periodogram(log_rain, log="no")
```



b. Use spec() function to produce a smoothe spectral density estimate. Please make sure you tried with different span sizes. One rule of thumb choice of span size is the square root of series length

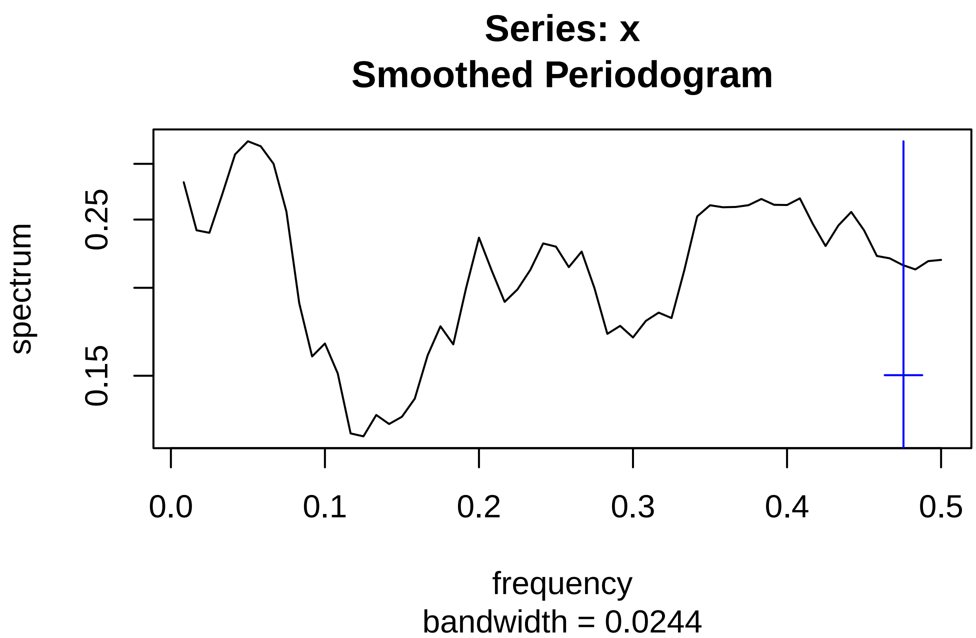
span size = 15

```
span <- 20
spec0 <- spectrum(log_rain, spans=span, plot=TRUE, detrend=TRUE)
```



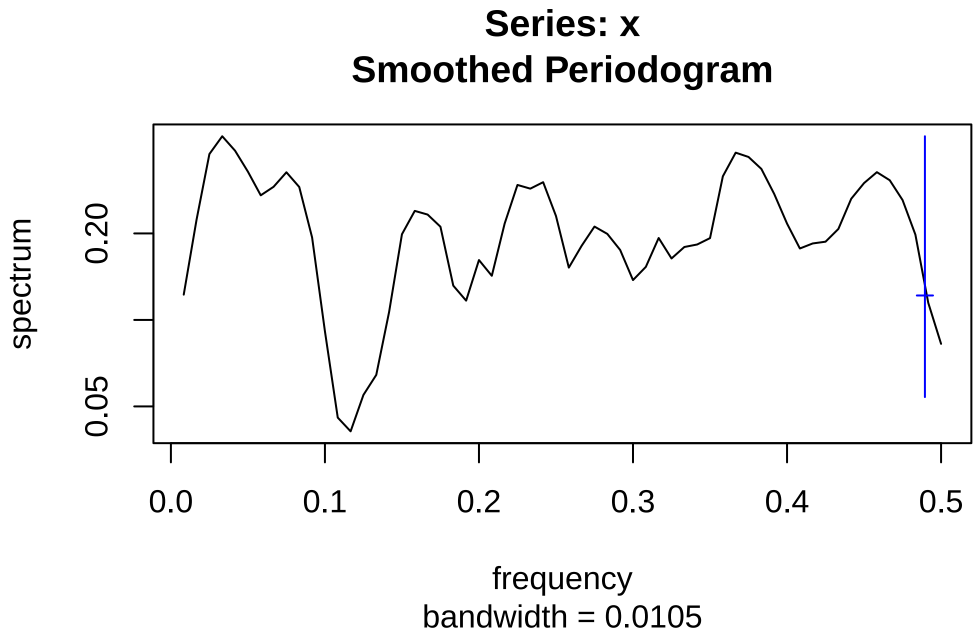
span size = sqrt(length(log_rain))

```
span <- sqrt(length(log_rain))
spec1 <-spectrum(log_rain, spans=span, plot=TRUE, detrend=TRUE)
```



span size = 5

```
span <- 5
spec2 <- spectrum(log_rain, spans=span, plot=TRUE, detrend=TRUE)
```



c. comment on the estimated spectrum

The estimated spectrum shows that there is a strong seasonality in our data, as it has many peaks and troughs. This would indicate that the rainfall is not random, but rather follows a pattern. We can see at the larger spans (20) that the data has becomed to smooth, and we cannot pickout the frequencies, with the lower span (5) we can see to many peaks and variance, the span of sqrt(length(log_rain)) seems to be the best fit, as we can see the frequencies but we aren't introducing to much variance.