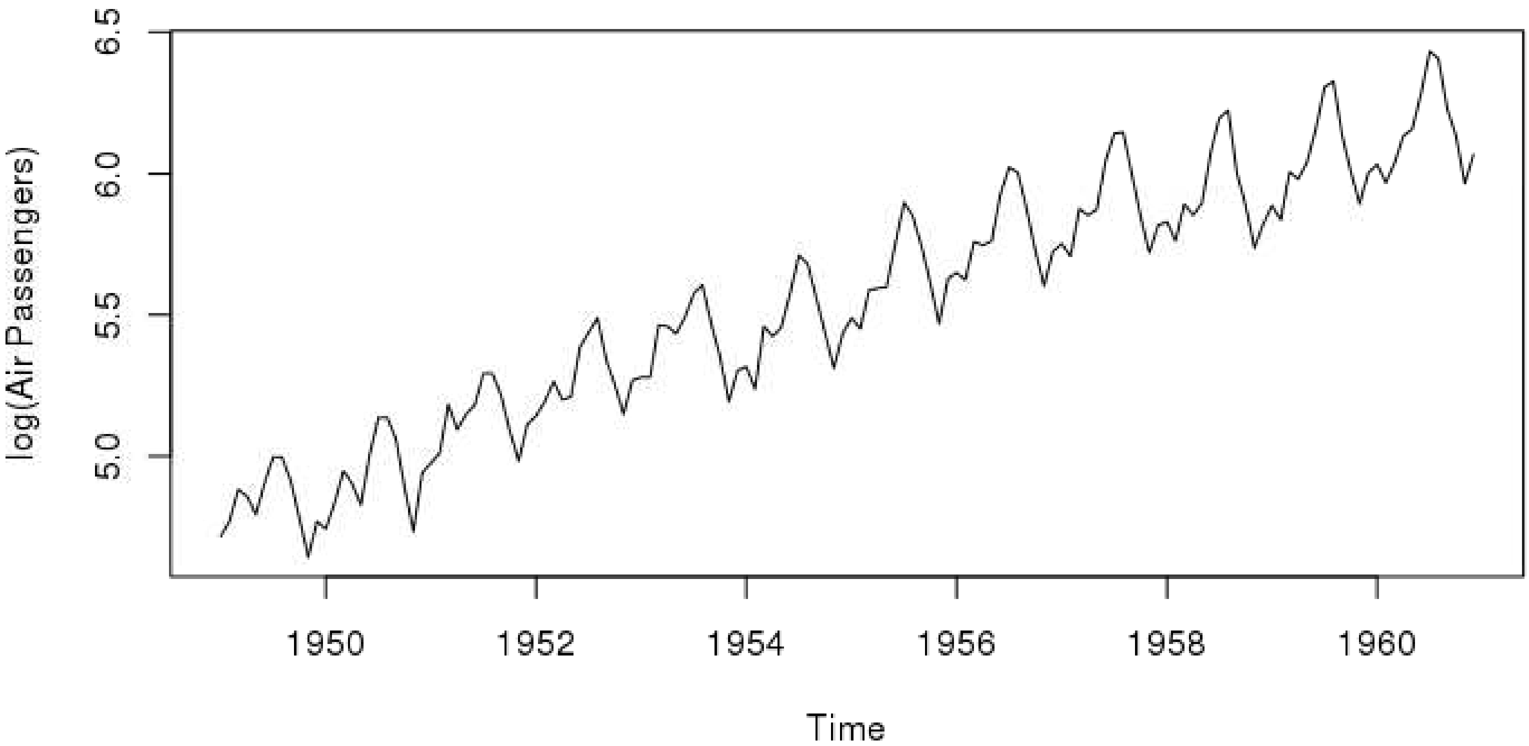
Spectral analysis

Beáta Stehlíková

Time series analysis

Motivation

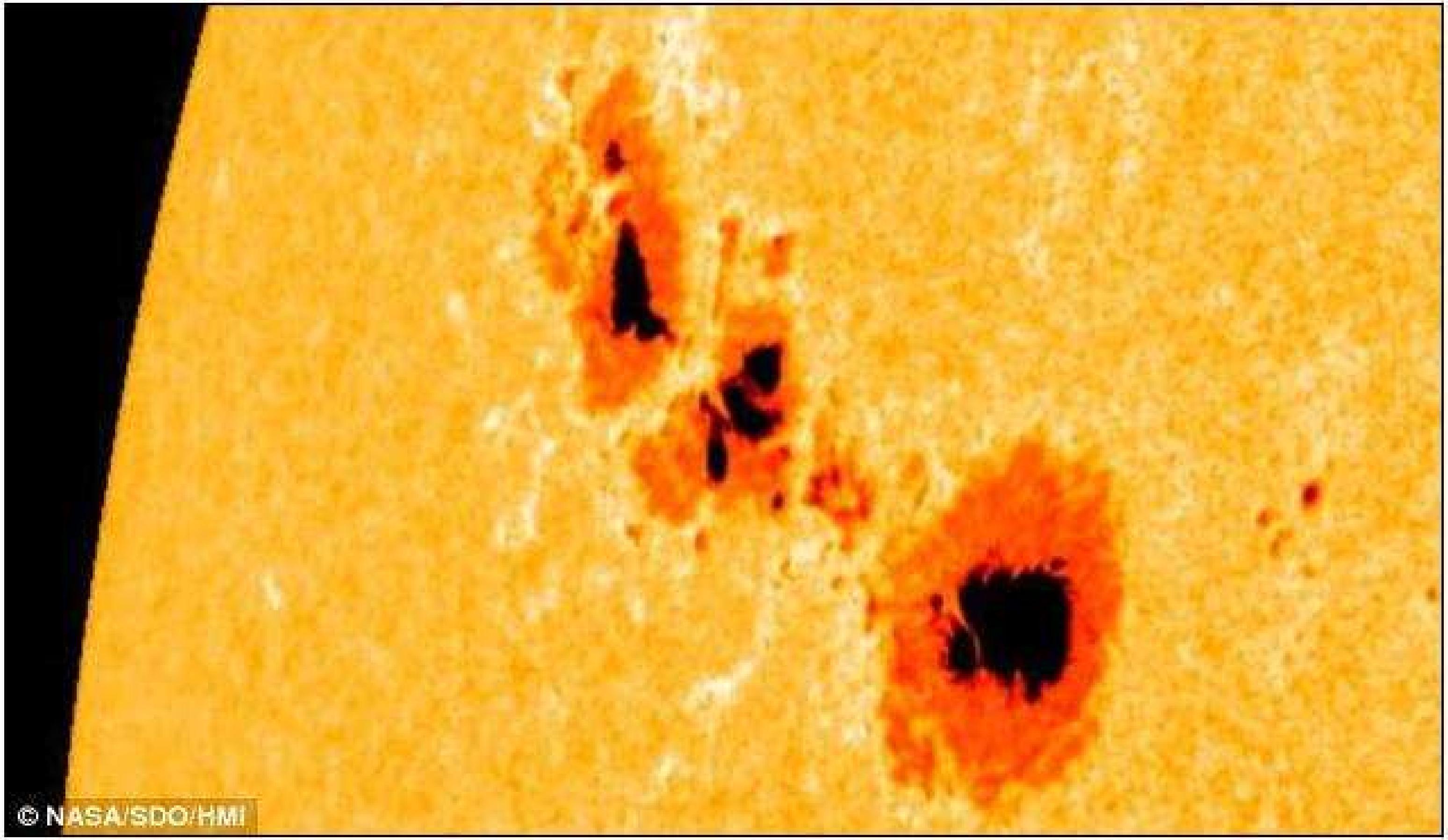
- Example about airline passengers



- Similarly e.g. monthly unemplyoment data, quartely GDP, ... $\rightarrow$ we have seen SARIMA models
- But not always we can deduce a clear period

Motivation

- Sunspots:

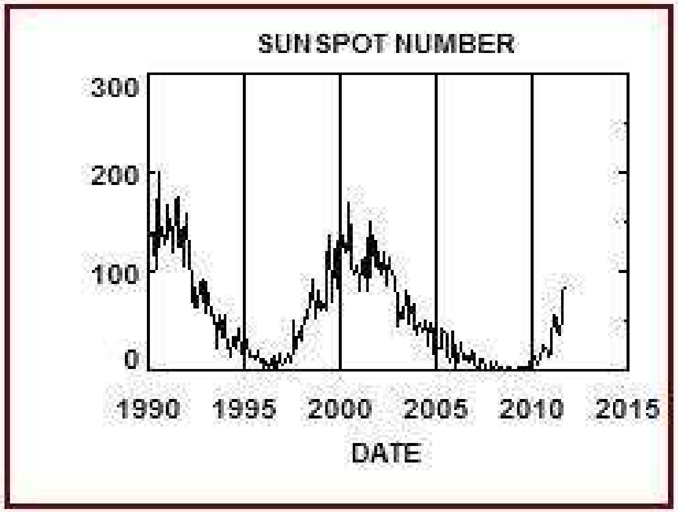


http://www.dailymail.co.uk/sciencetech/article-2042428/Best-auroras-seen-Britain-thanks-huge-solar-flares.html

Motivation

- Sunspots:

Sunspot Numbers



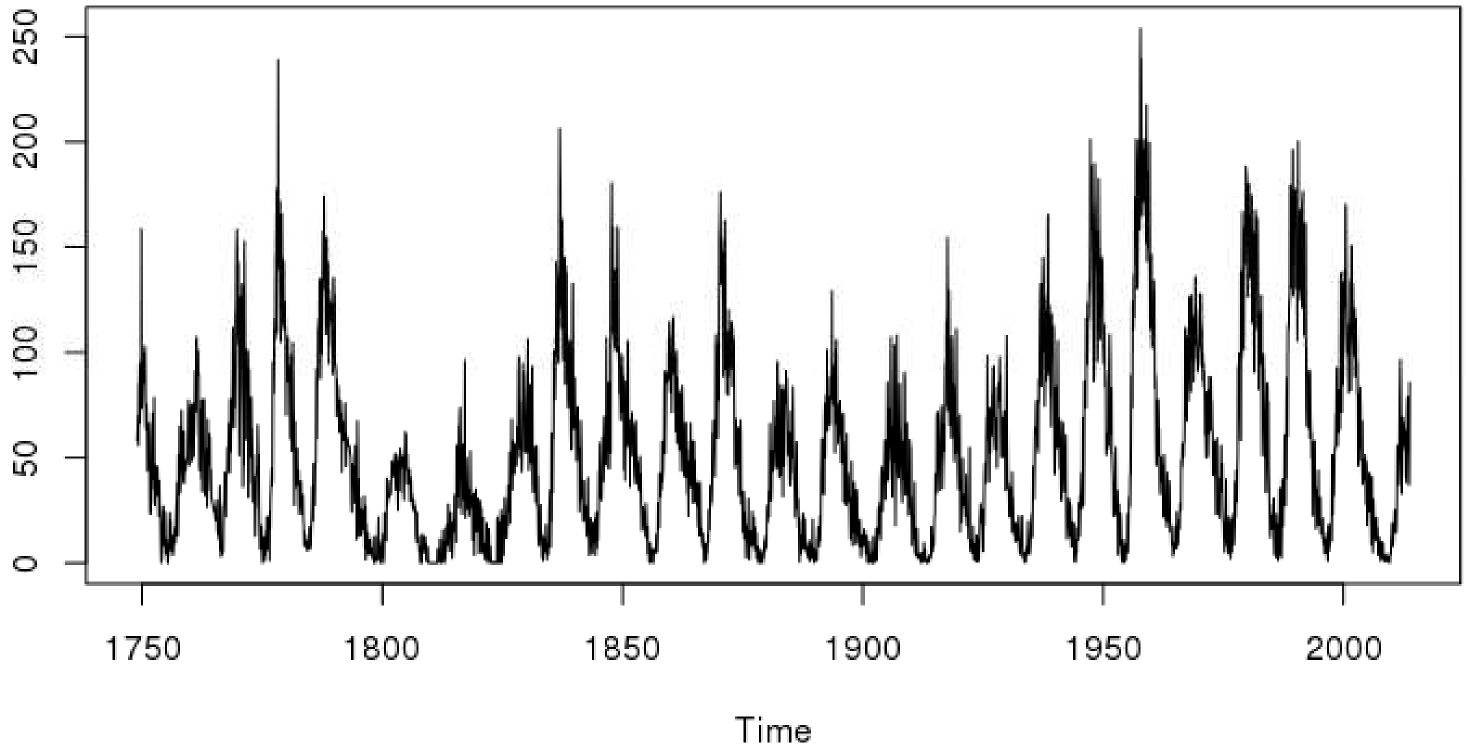
Click on image for larger version.

In 1610, shortly after viewing the sun with his new telescope, Galileo Galilei (or was it Thomas Harriot?) made the first European observations of Sunspots. Continuous daily observations were started at the Zurich Observatory in 1849 and earlier observations have been used to extend the records back to 1610. The sunspot number is calculated by first counting the number of sunspot groups and then the number of individual sunspots.

http://solarscience.msfc.nasa.gov/SunspotCycle.shtml

Motivation

- Sunspots - a longer time series:



Data from the website http://solarscience.msfc.nasa.gov/SunspotCycle.shtml

- Question: How to determine the period?

Spectrum

- Sequence $\{\gamma_j\}_{j=-\infty}^{\infty} \rightarrow$ generating function

$$g(z) = \sum_{j=-\infty}^{\infty} \gamma_j z^j$$

- Stationary process Y with autocovariances $\{\gamma_j\}_{j=-\infty}^{\infty}$ $\rightarrow$ spectrum

$$s_Y(\omega) = \frac{1}{2\pi} g(e^{-i\omega}) = \frac{1}{2\pi} \sum_{j=-\infty}^{\infty} \gamma_j e^{-i\omega j},$$

where i is imaginary unit.

Properties of spectrum

- Spectrum $s_Y(\omega)$:
  - can be written as:

$$s_Y(\omega) = \frac{1}{2\pi}\left[\gamma_0 + 2\sum_{j=1}^{\infty}\gamma_j \cos(\omega j)\right]$$

  - has real values
  - is an even function
  - has a period 2π

$\rightarrow$ it is sufficient to know the values between 0 and π

- It can be proved that $s_Y(\omega) \geq 0$ [Fuller, 1976]

Spectrum and variance

- Calculation of autocovariances from the spectrum:

$$\gamma_k = \int_{-\pi}^{\pi} s_Y(\omega) e^{i\omega k} d\omega$$

- For $k = 0$ we get: $\gamma_0 = \int_{-\pi}^{\pi} s_Y(\omega) d\omega$, and since the spectrum is an even function:

$$\gamma_0 = 2 \int_0^{\pi} s_Y(\omega) d\omega,$$

  so the variance γ_0 is twice the area below the graph of the spectrum on the interval $[0, \pi]$
- So from the behaviour of the spectrum we can see which frequencies add the most to the variance of the process - those, for which the spectrum has a high value

Estimation: sample periodogram

- We have data $y_1, \ldots y_T \rightarrow$ and we want to estimate the spectrum of the time serie
- First idea: we replace the autocovariances in the definition with their estimates, in this way we get - sample periodogram:

$$\hat{s}_y(\omega) = \frac{1}{2\pi} \sum_{j=-T+1}^{T-1} \hat{\gamma}_j e^{-i\omega j} = \frac{1}{2\pi} \left[\hat{\gamma}_0 + 2 \sum_{j=1}^{T-1} \hat{\gamma}_j \cos(\omega j) \right]$$

- Problems:
  - ◇ estimates have a high variance
  - ◇ accuracy does not get better whe we have more data (because we are estimating more autocovariances)

$\rightarrow$ we need another estimation of the spectrum

Estimation: sample periodogram

- Problems - in more detail:
  - [Fuller, 1976]: for a large sample size, the ratio $\frac{2\hat{s}_y(\omega)}{s_Y(\omega)}$ has approximately $\chi^2(2)$ dispribution and these ratios re for different ω approximately independent
  - $E[\chi^2(2)] = 2$, and hence

$$E[\hat{s}_y(\omega)] \sim s_Y(\omega)$$

    - that is OK
  - But 95 percent confidence interval for $\chi^2(2)$ is $(0.05, 7.4)$ and hence CI for the spectrum is

$$(0.025\hat{s}_y(\omega), 3.7\hat{s}_y(\omega))$$

    - too wide

Estimating spectrum - a better estimate

- Idea: if the frequencies are close, also the spectrum values are close $\to$ as an estimate of the spectrum corresponding to a given frequency we take a weighted average of sample periodogram values $\hat{s}_y$ for neighbouring frequencies:

$$\hat{s}_Y(\omega_j) = \sum_{m=-h}^{h} \kappa(\omega_{j+m}, \omega_j)\hat{s}_y(\omega_{j+m}) \qquad (1)$$

where
  - $\omega_j = 2\pi j/T$
  - constant h gives the number of neighbouring frequencies which we take into account when computing the estimate (called bandwidth)
  - function κ determines weights, for these frequencies (they sum to 1)

Example 1: simulated data

- Recall AR(2) process with periodic character:

$$x_t = 1.4x_{t-1} - 0.85x_{t-2} + u_t$$

- We know:
  - ◇ correlation satisfy the difference equation

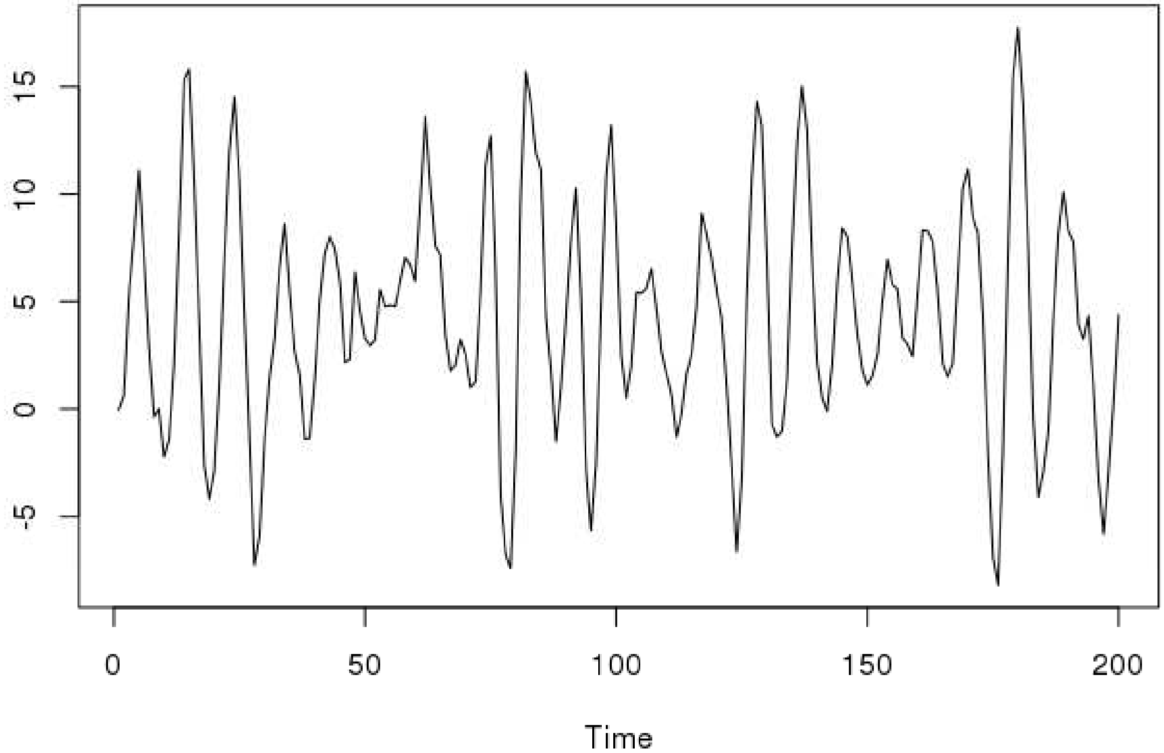
    $$\rho(t) - 1.4\rho(t-1) + 0.85\rho(t-2) = 0,$$

    which has a general solution

    $$\rho(t) = 0.922^t(c_1 \cos(0.709t) + c_2 \sin(0.709t))$$
  - ◇ sine and cosine in a general solution $\cos(kt), \sin(kt)$ $\rightarrow$ period $\frac{2\pi}{k} = 8.862 \approx 9$

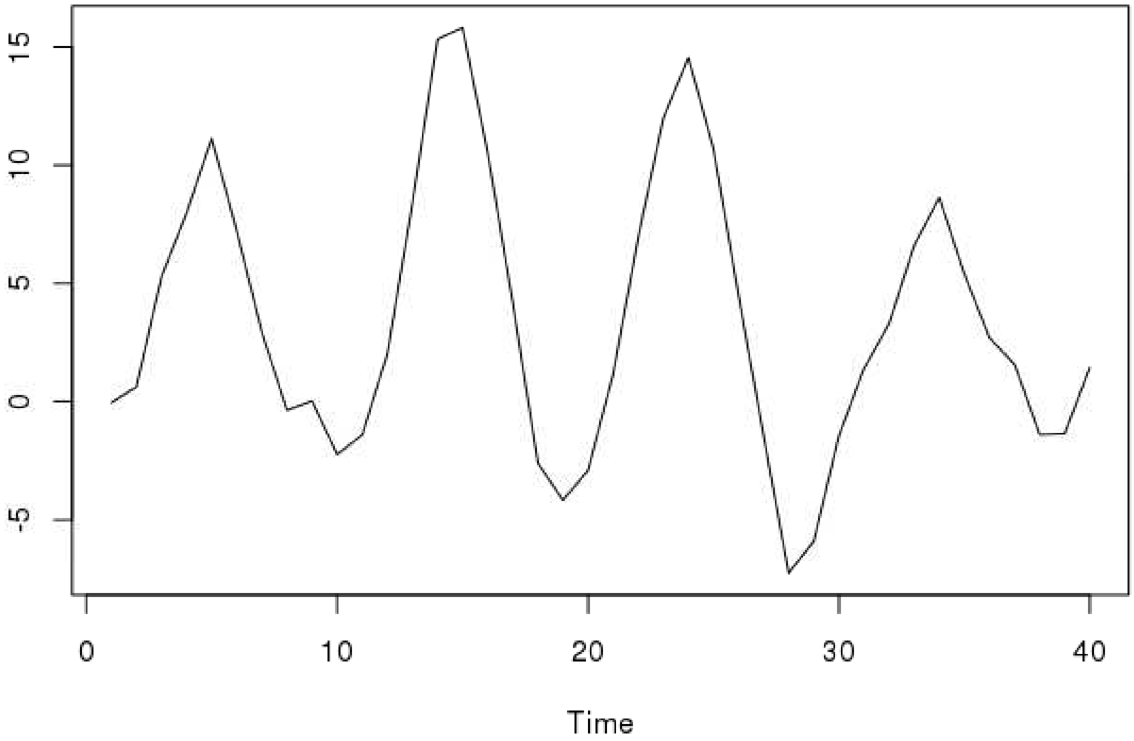
Example 1: simulated data

- Sample trajectory:



Example 1: simulated data

- Simulated frajectory - a shorter time interval (we can see the period better):



Example 1: simulated data

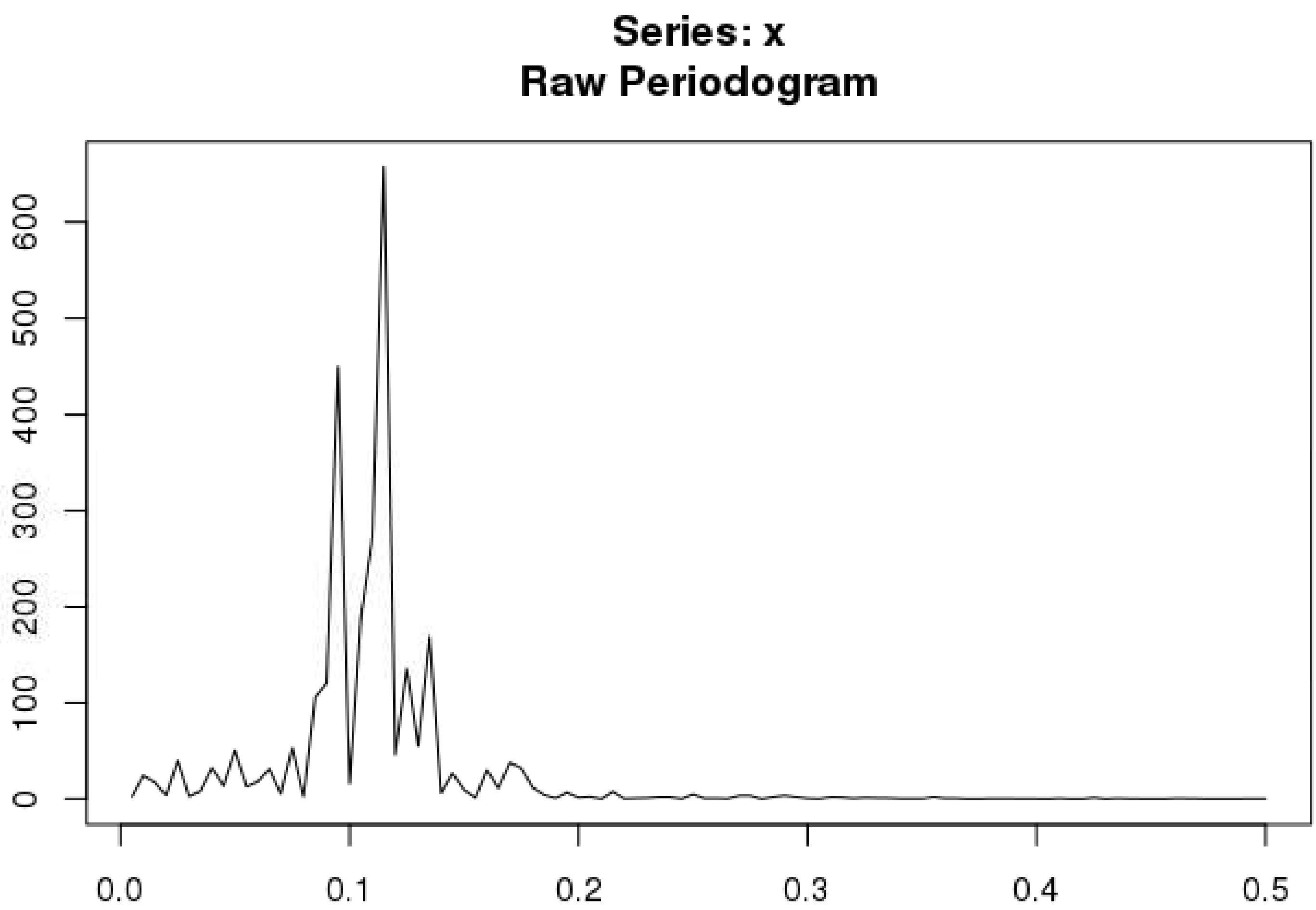
- Estimating the spectrum in R:
  - sample periodogram:

    ```
    spectrum(x, log="no")
    ```
  - smoothed:

    ```
    spectrum(x, kernel("daniell"), log="no")
    spectrum(x, kernel("modified.daniell"), log="no")
    ```
  - different scaling of x-axis: from 0 to 1/2, so period = 1/frequency)

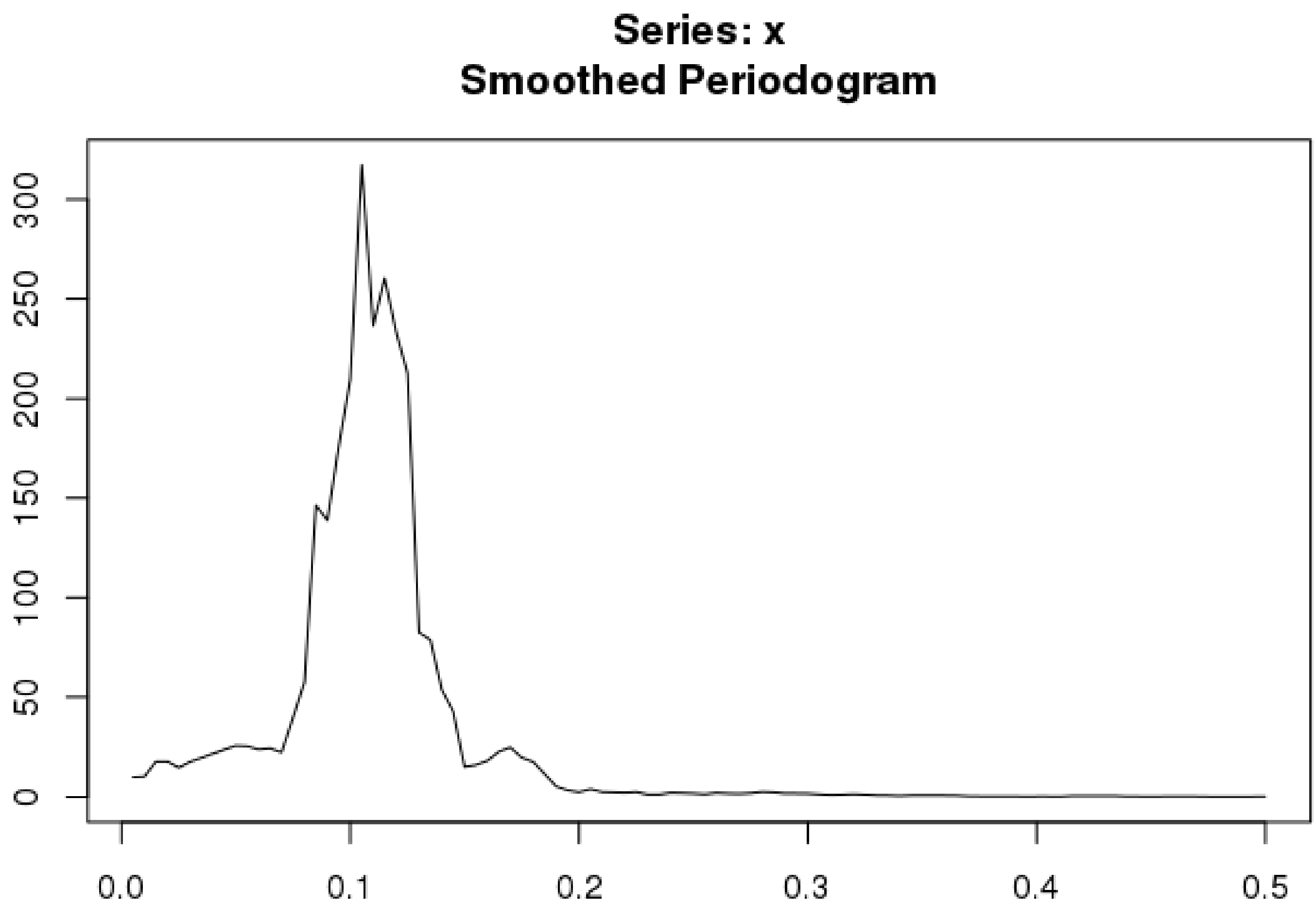
Example 1: simulated data

- Sample periodogram: spectrum(x, log="no")



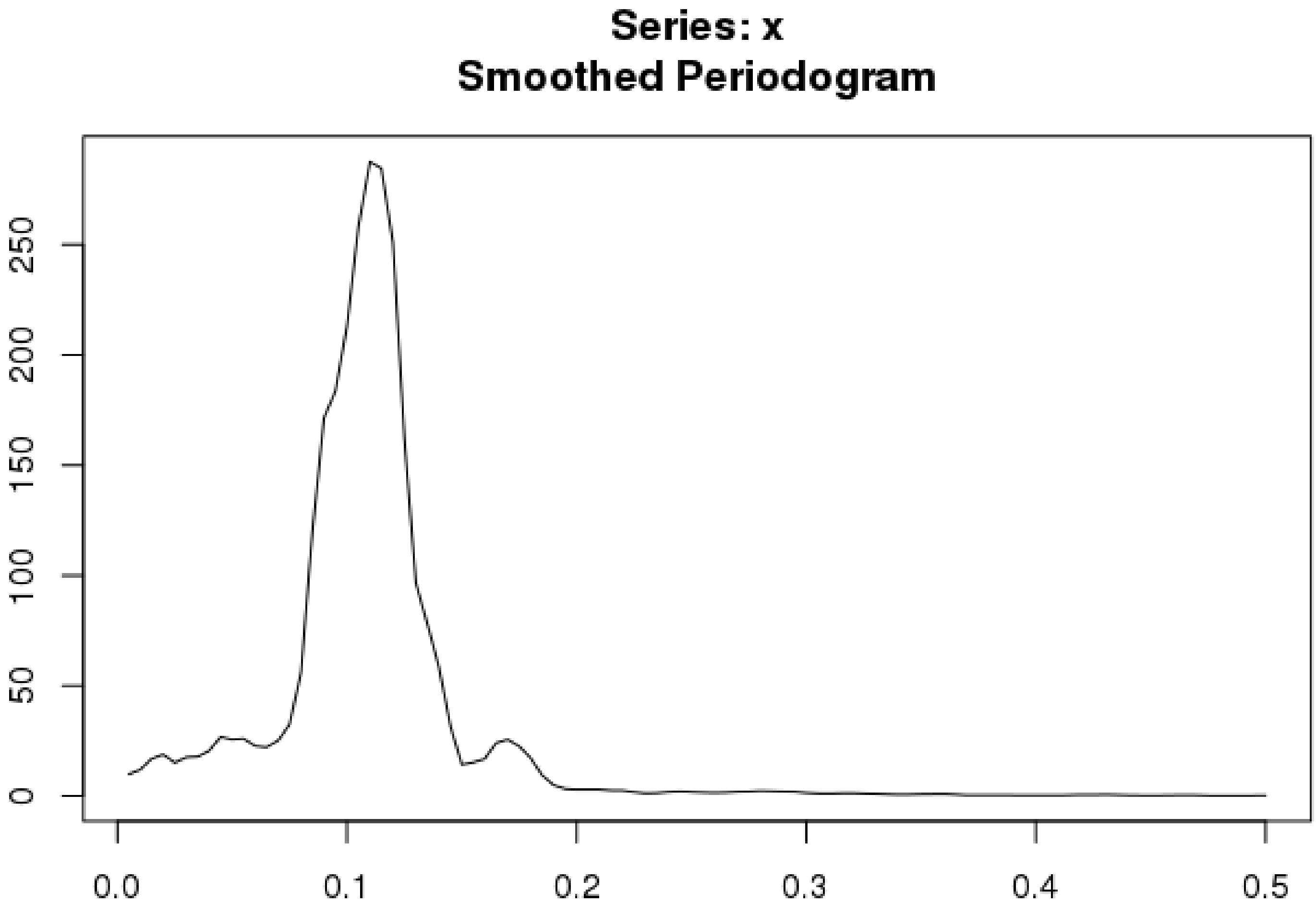
Example 1: simulated data

- Smoothed sample periodogram:
  spectrum(x, kernel("daniell"), log="no")



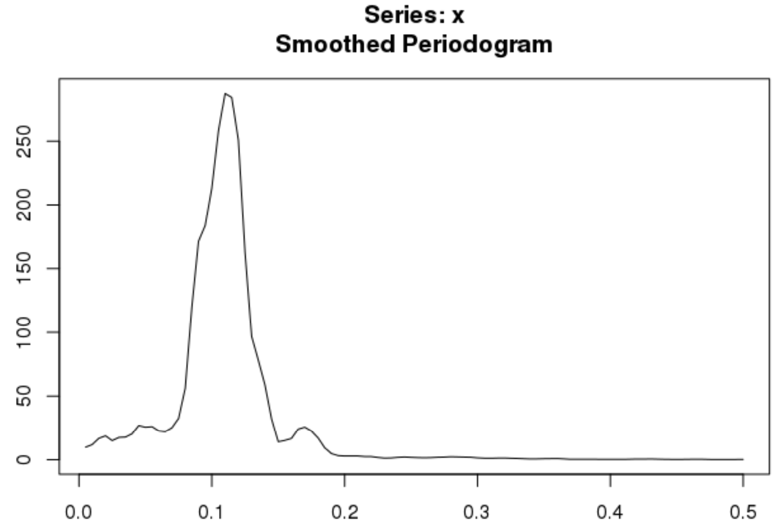
Example 1: simulated data

- Smoothed sample periodogram:
  spectrum(x, kernel("modified.daniell"), log="no")

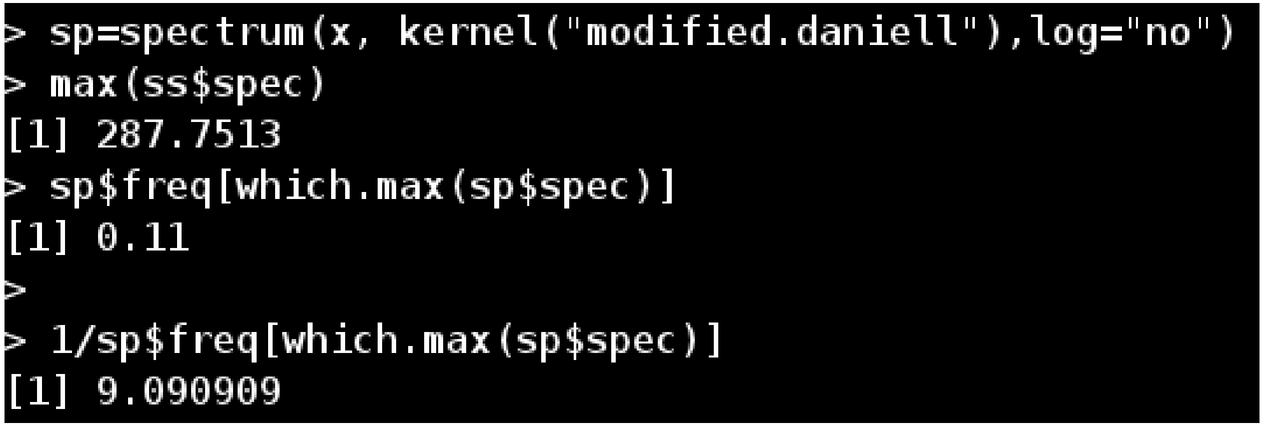


Example 1: simulated data

- Recall:



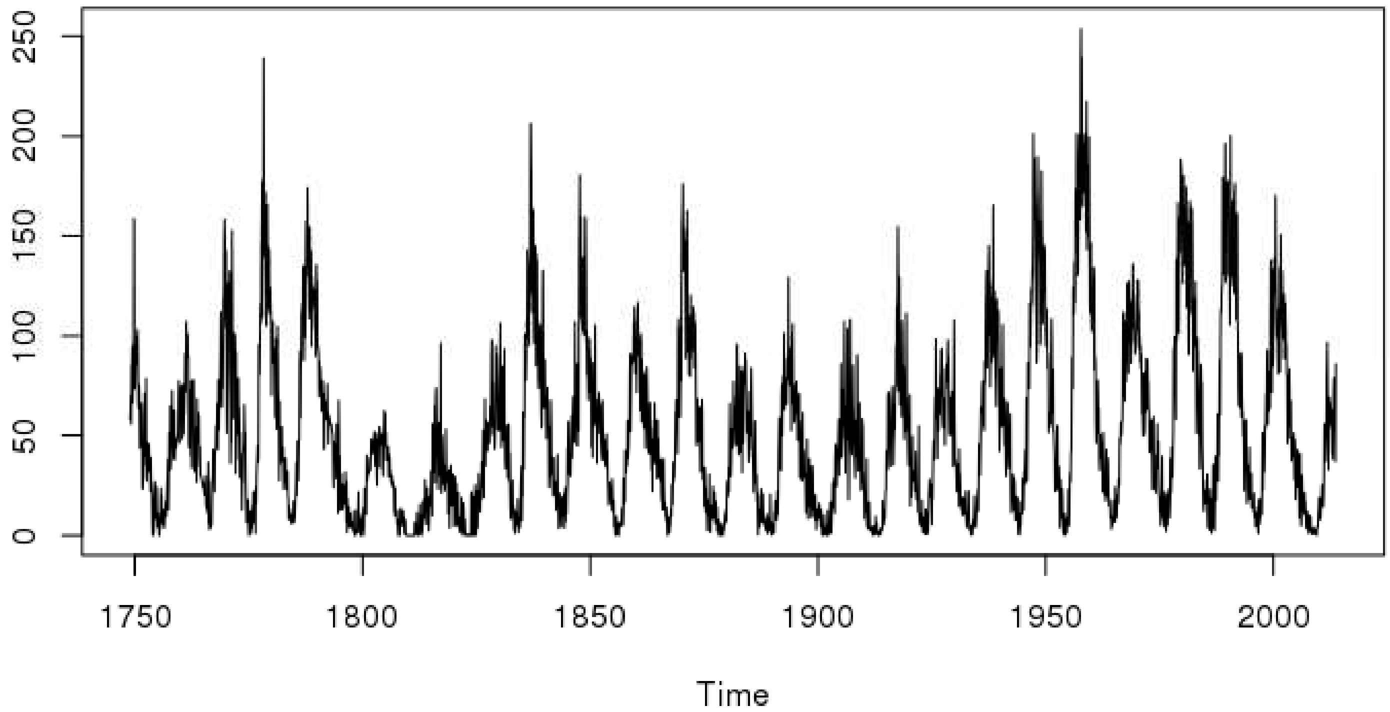
- We find maximum and corresponding period:



```
> sp=spectrum(x, kernel("modified.daniell"),log="no")
> max(ss$spec)
[1] 287.7513
> sp$freq[which.max(sp$spec)]
[1] 0.11
>
> 1/sp$freq[which.max(sp$spec)]
[1] 9.090909
```

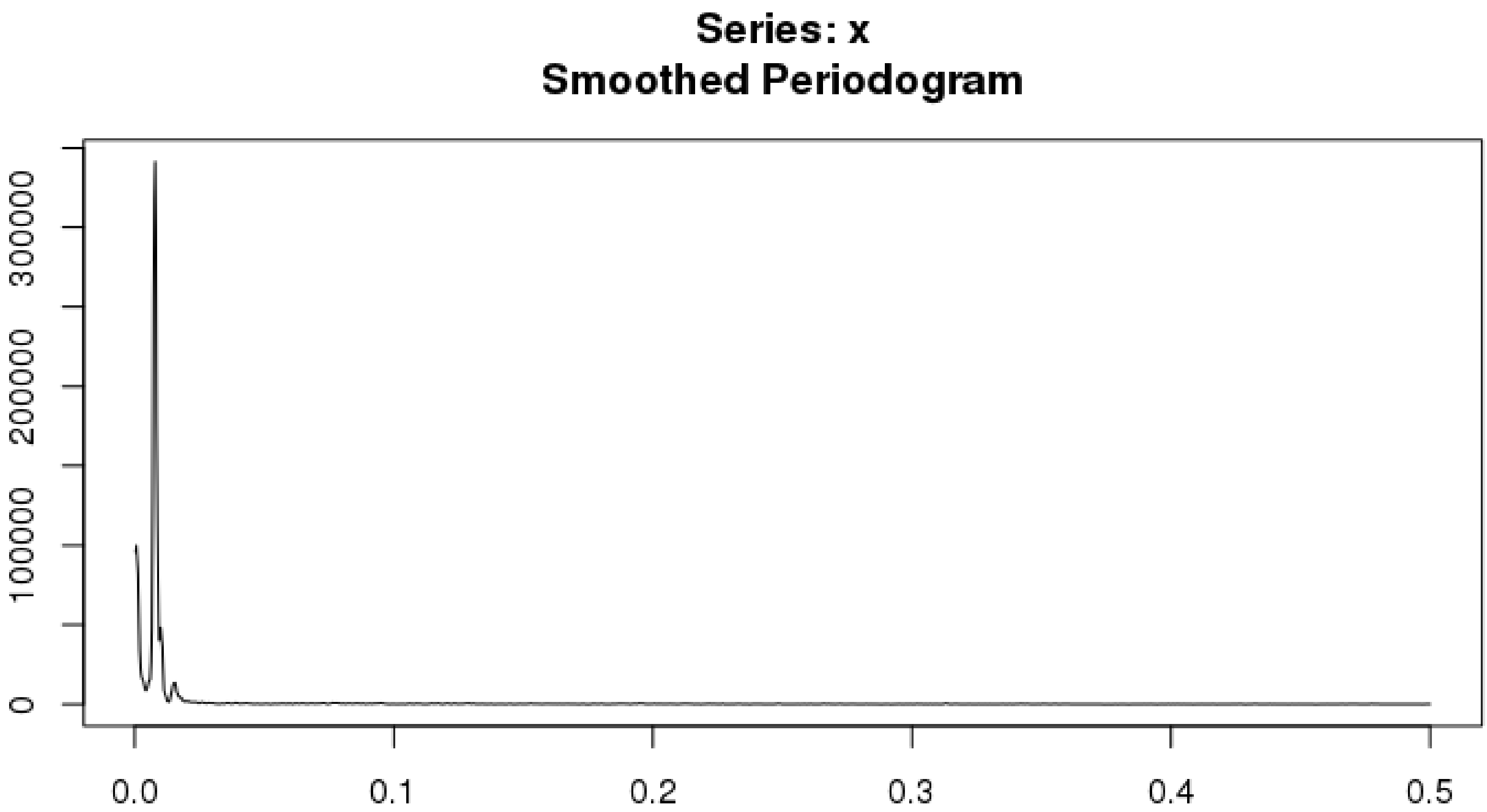
Example 2: sunspots

- Data:
  sun <- read.table("sun.txt")
  plot(ts(sun),frequency=12,start=c(1749,1))



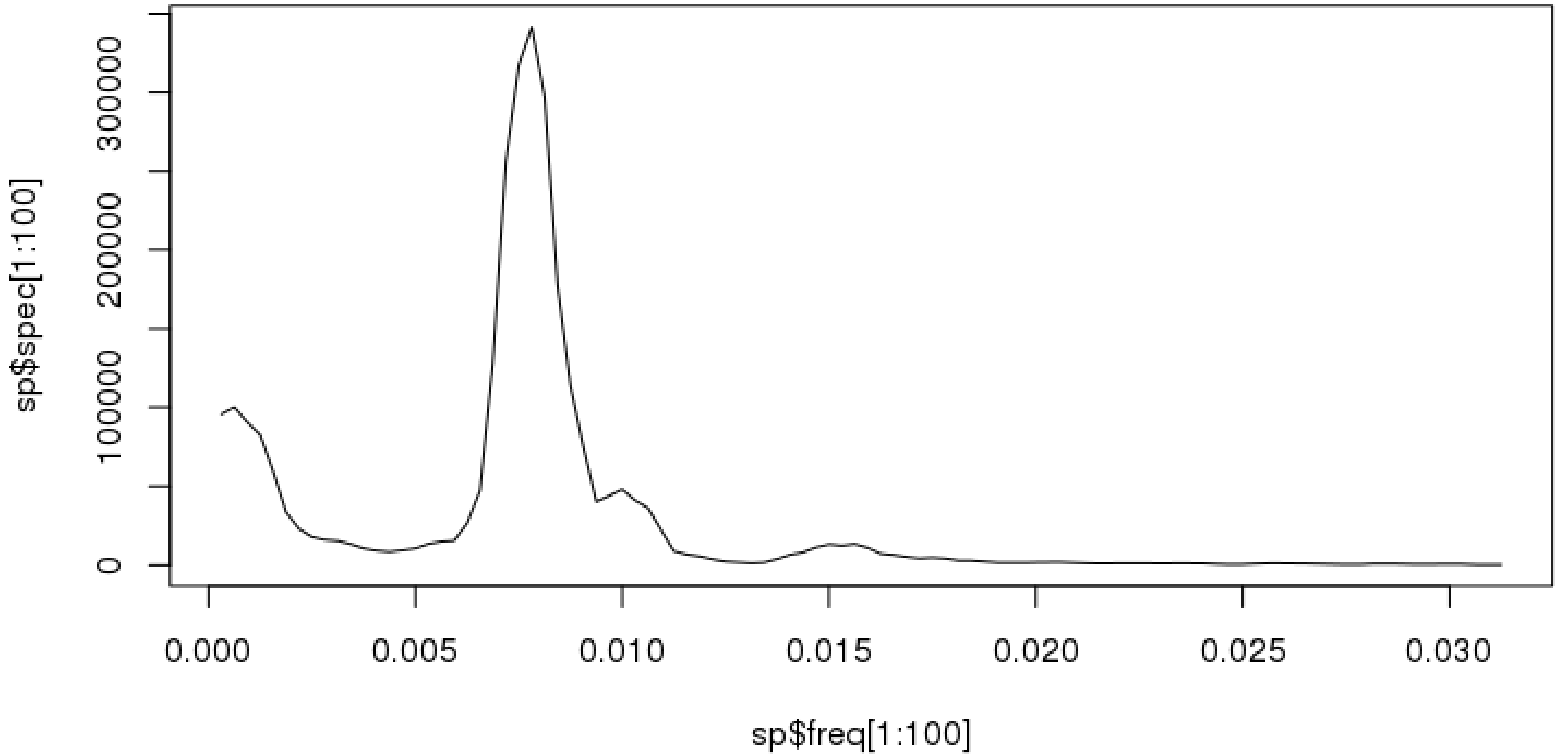
Example 2: sunspots

- Estimate of the spectrum:
  spectrum(sun, kernel("modified.daniell"), log="no")



Example 2: sunspots

- Zoom:
  plot(sp$freq[1:100], sp$spec[1:100], type="l")



Example 2: sunspots

- We find the maximum and corresponding period:

```
> 1/sp$freq[which.max(sp$spec)]
[1] 128
> 1/sp$freq[which.max(sp$spec)]/12
[1] 10.66667
```

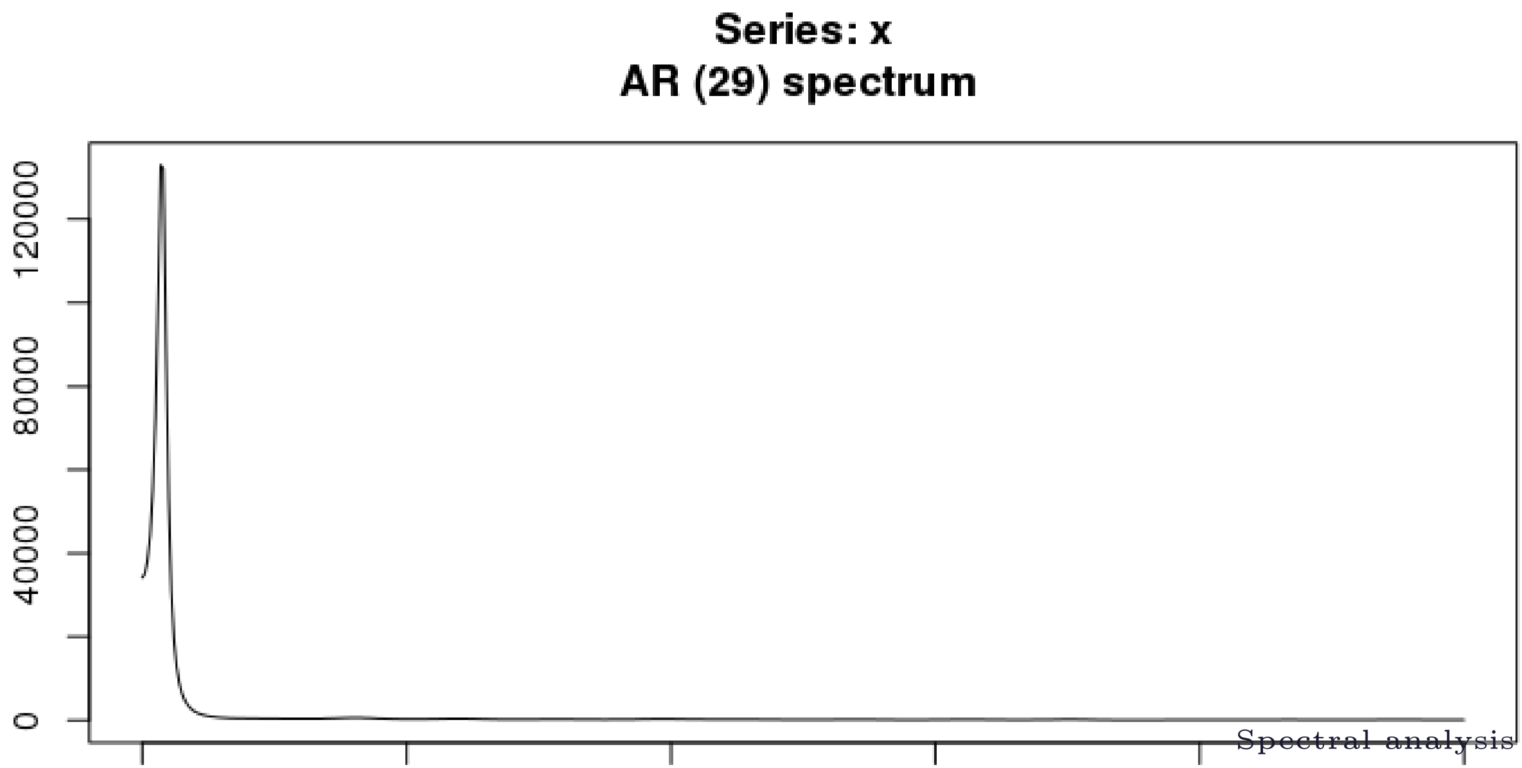
- From the website:

> The "sunspot number" is then given by the sum of the number of individual sunspots and ten times the number of groups. Since most sunspot groups have, on average, about ten spots, this formula for counting sunspots gives reliable numbers even when the observing conditions are less than ideal and small spots are hard to see. Monthly averages (updated monthly) of the sunspot numbers (181 kb JPEG image), (307 kb pdf-file), (62 kb text file) show that the number of sunspots visible on the sun waxes and wanes with an approximate 11-year cycle.

http://solarscience.msfc.nasa.gov/SunspotCycle.shtml

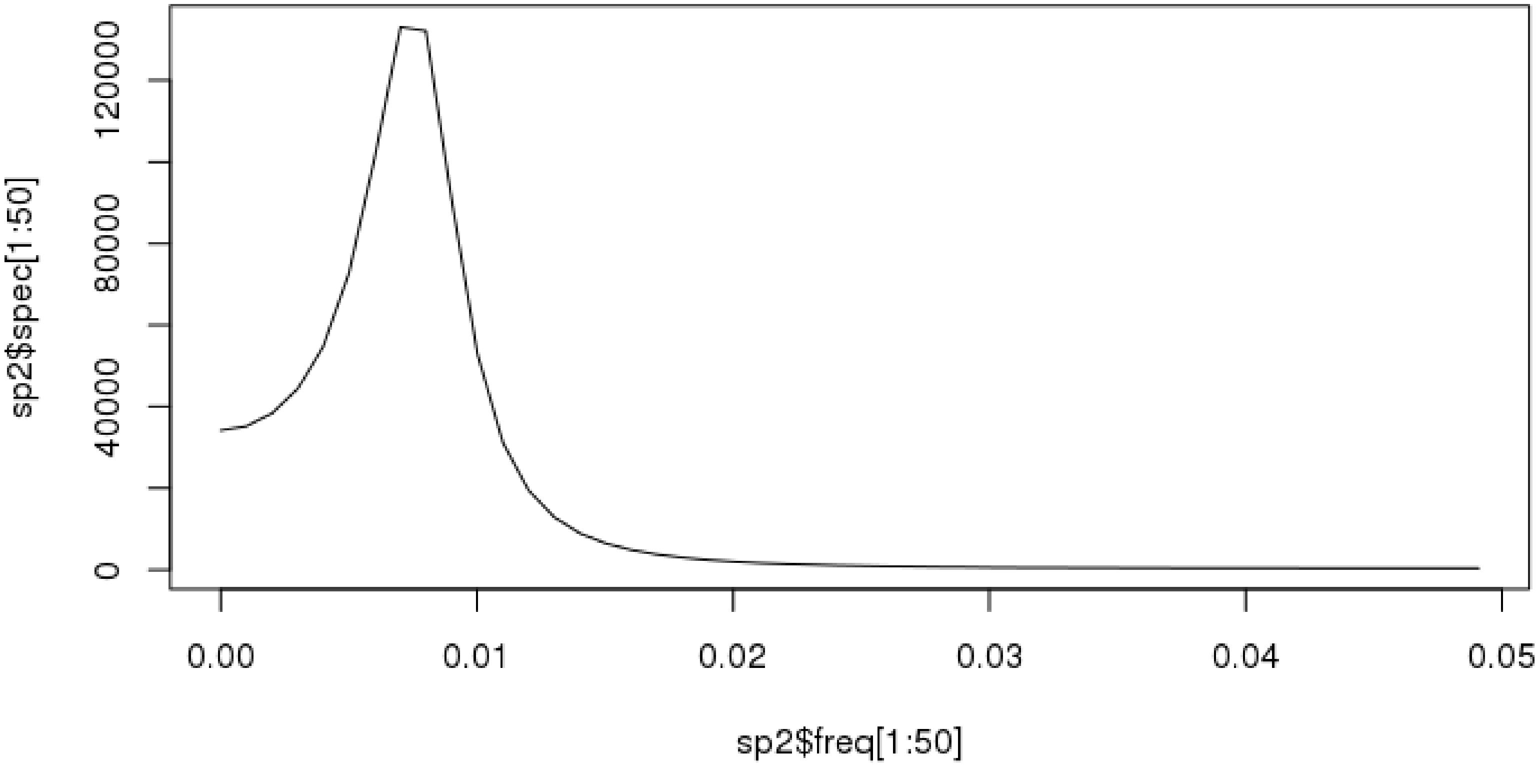
Example 3: estimation using AR models

- We model data as an AR process
- Spectrum is then estimated as the spectrum of that AR process
- Sunspots data: sp2<-spectrum(ts(sun), method="ar", log="no")



Example 3: estimation using AR models

- Zoom:



Example 3: estimation using AR models

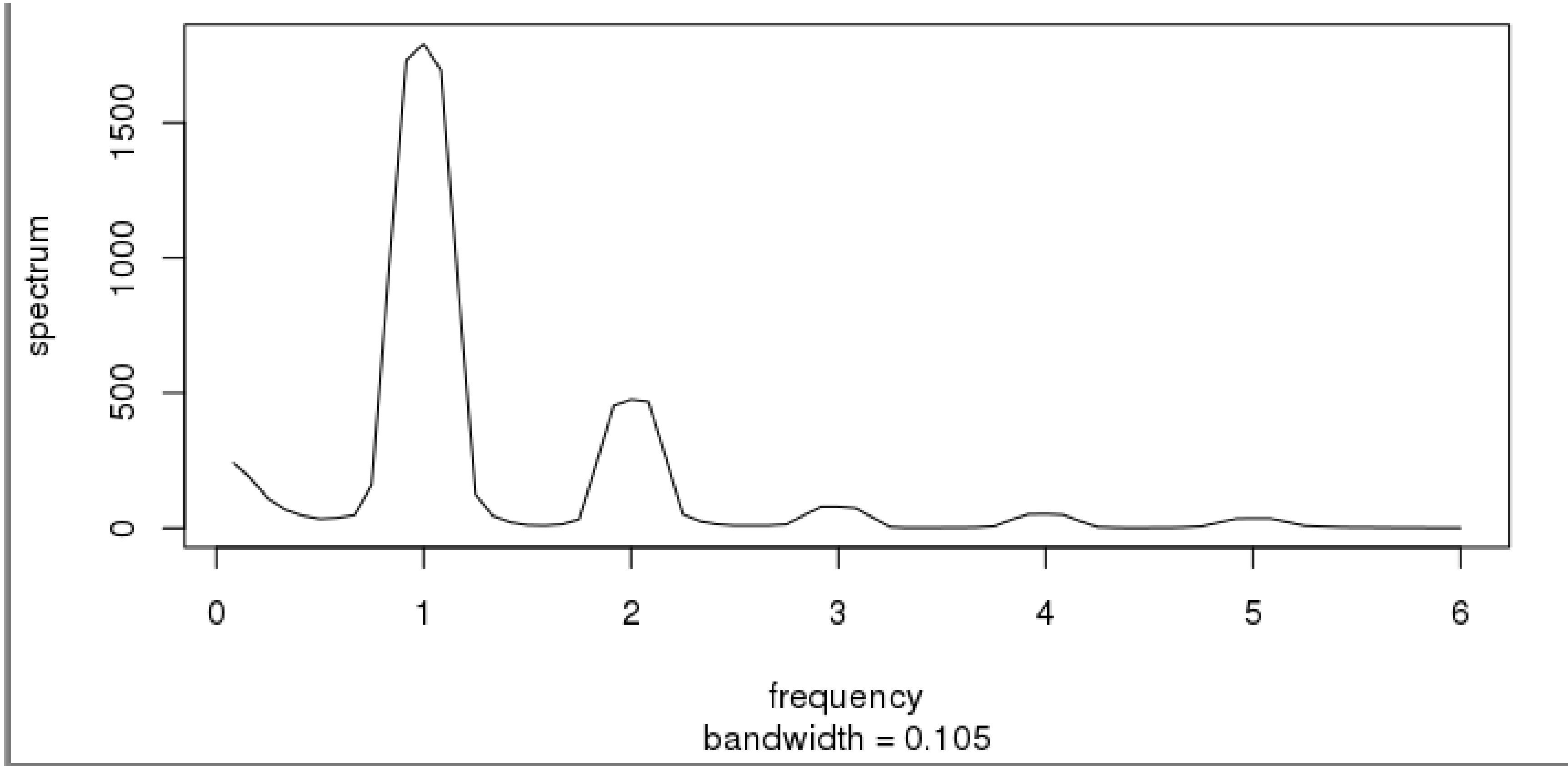
- Finding maximum:

```
> sp2=spectrum(ts(sun),method="ar",log="no")
>
> 1/sp2$freq[which.max(sp2$spec)]
[1] 142.5714
> 1/sp2$freq[which.max(sp2$spec)]/12
[1] 11.88095
```

Notes on R

- In the first approach (smoothed sample periodogram):
  - Data are detrended (linear trend).
  - If the data have a time structure, it is kept when computing frequencies and periods.
- EXAMPLE: airline passengers
  - in R:
    ```
    data(AirPassengers)
    x<-AirPassengers
    ```
  - time structure: unit of time = year, frequency of the data = 12

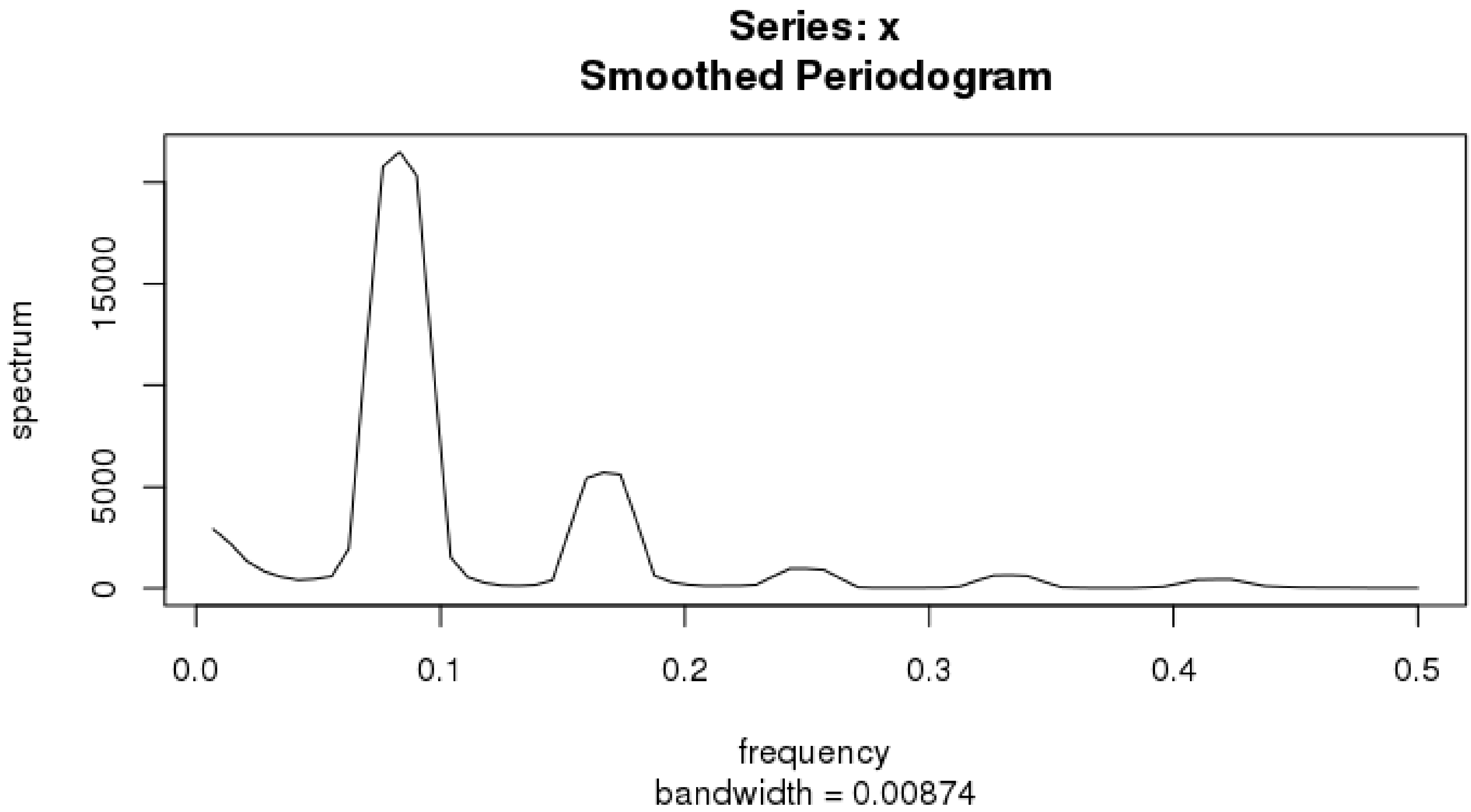
Notes on R




```
> data(AirPassengers)
> x=AirPassengers
> sp=spectrum(x,kernel("modified.daniell"),log="no")
> 1/sp$freq[which.max(sp$spec)]
[1] 1
```

Notes on R

For a comparison, without the time structure:




```
> sp=spectrum(ts(x,frequency=1),kernel("modified.daniell"),log="no")
> 1/sp$freq[which.max(sp$spec)]
[1] 12
```