Bruce Campell NCSU ST 534 Exam 2

17 November, 2017

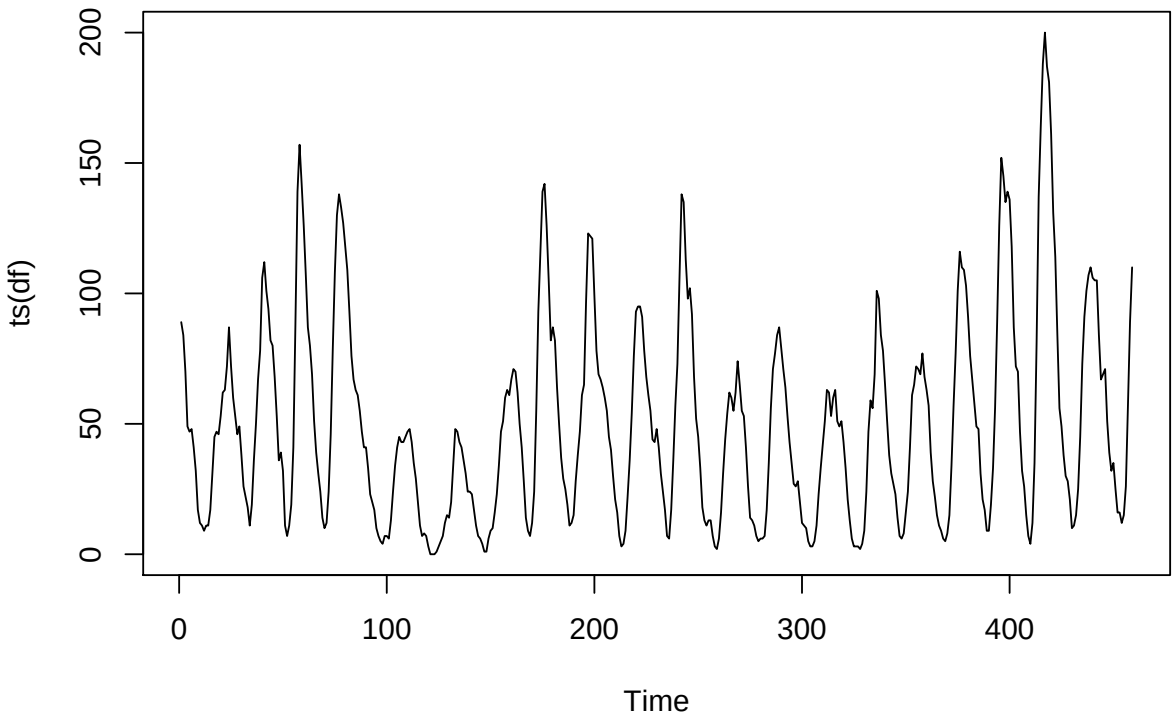
---

1 sunspotz analyis

Consider the series sunspotz in the package astsa and answer the following questions:

(a) Plot sunspotz. Comment on the notable features of the time series?

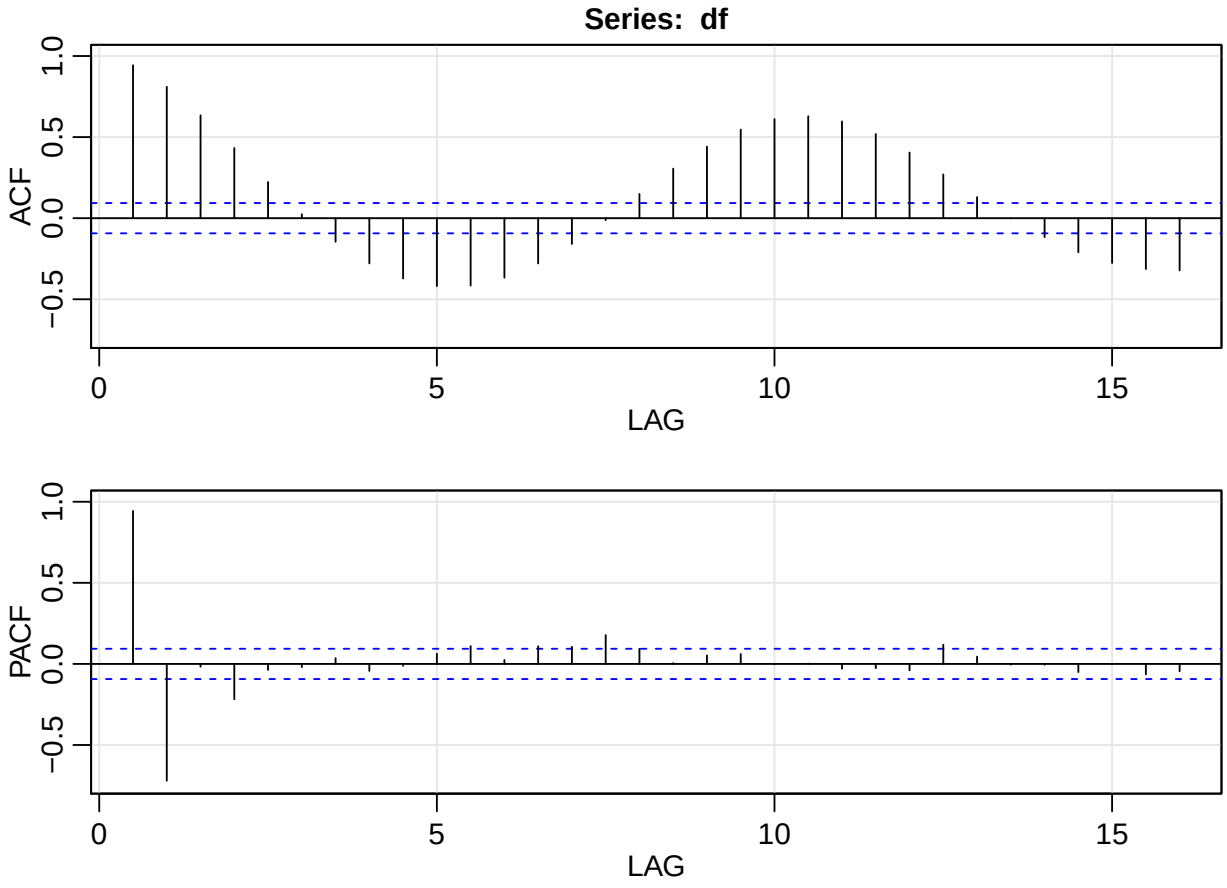
```
rm(list = ls())
library(astsa)
data(sunspotz, package = "astsa")
df <- sunspotz
plot(ts(df))
```



From the documentation sunspotz. is the *biannual smoothed (12-month moving average) number of sunspots from June 1749 to December 1978; n = 459.* The format is `Time Series: Start = c(1749, 1) End = c(1978, 1) Frequency = 2` There appear to be two or more fundamental frequencies in the series. We note the appearance of short term and long term cyclical behavior. In regards to modelling with this data we would want to take into consideration that the period of observations is shore relative to the evolution of the dynamical system generating the data, and that the technology for making observations may have evolved over the period of data collection.

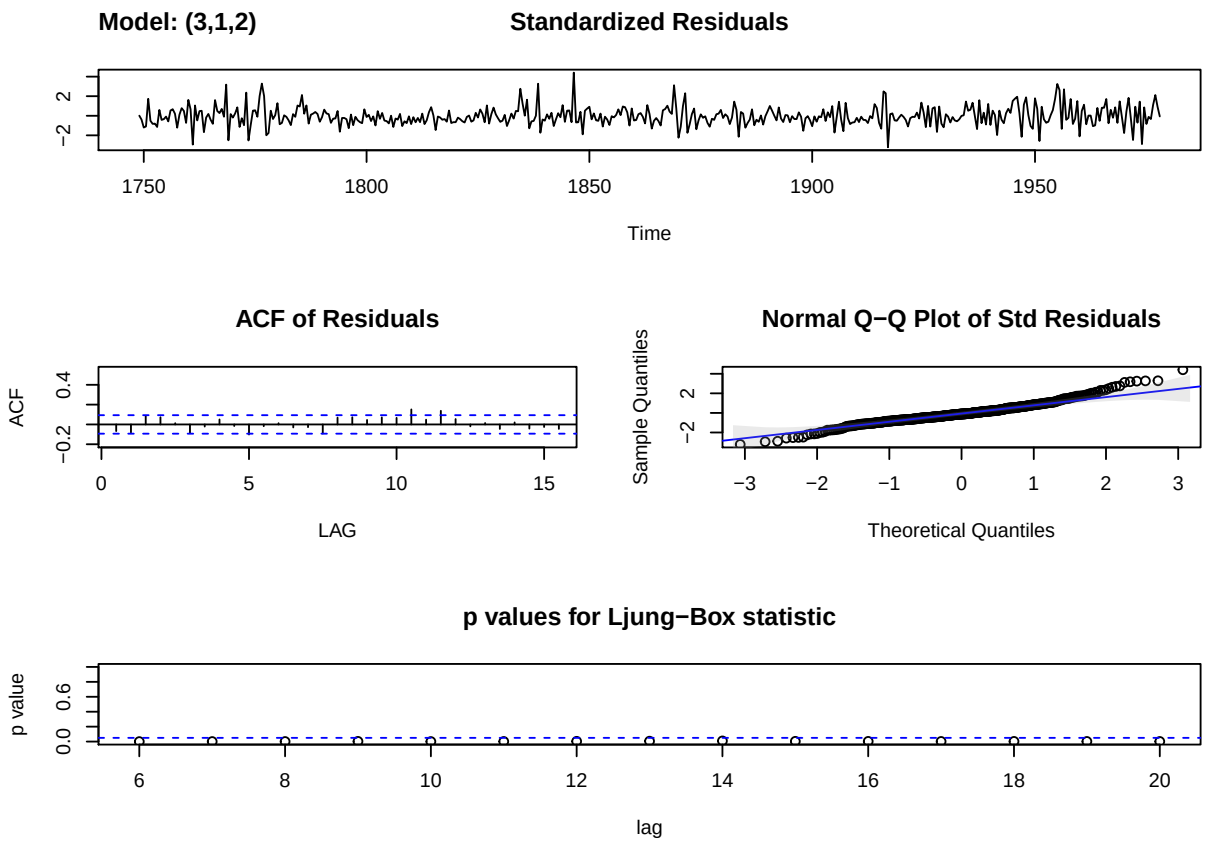
(b) Plot the ACF of sunspotz. Do you think it is appropriate to model this using a stationary ARMA process ?

```
invisible(acf2(df))
```



We note the presence of strong long range correlation. At first glance we would be hesitant to model this with a stationary ARMA process. We'd possibly perform a unit root test and consider modelling this data with an ARIMA process. For fun, we do some modelling in the temporal domain.

```
invisible(model <- sarima(df, 3, 1, 2))
```

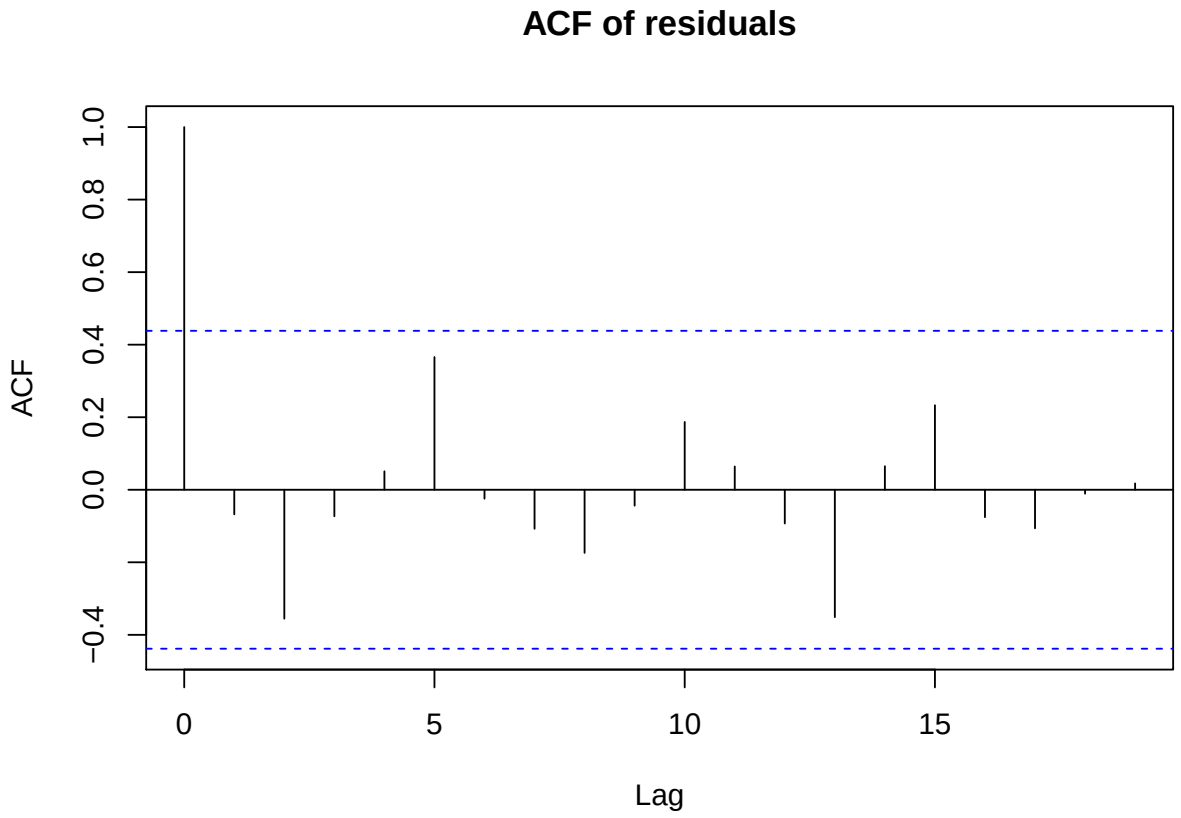



```
model$ttable
```

We note some outlier in the the standardized residuals, heavy tails in the QQ plot, and that the individual Ljung-Box p-values are all close to being significant. We also note a presence of a lag 11,12 acf component in the residuals. This is consistent with our initial observation that the data had short term and long term cyclic trends.

For more fun let's calculate the Ljung-Box-Pierce Q-statistic to check for systemic autocorrelation in the residuals.

```
n <- length(df)
H <- 20
r <- model$fit$residuals[1:H]
acf.residuals <- acf(r, H, main = "ACF of residuals")
```




```
sum.denominator <- n - seq(H, 1, by = -1)
r.s <- acf.residuals$acf^2/sum.denominator
Q <- n * (n + 2) * sum(r.s)
Q
```

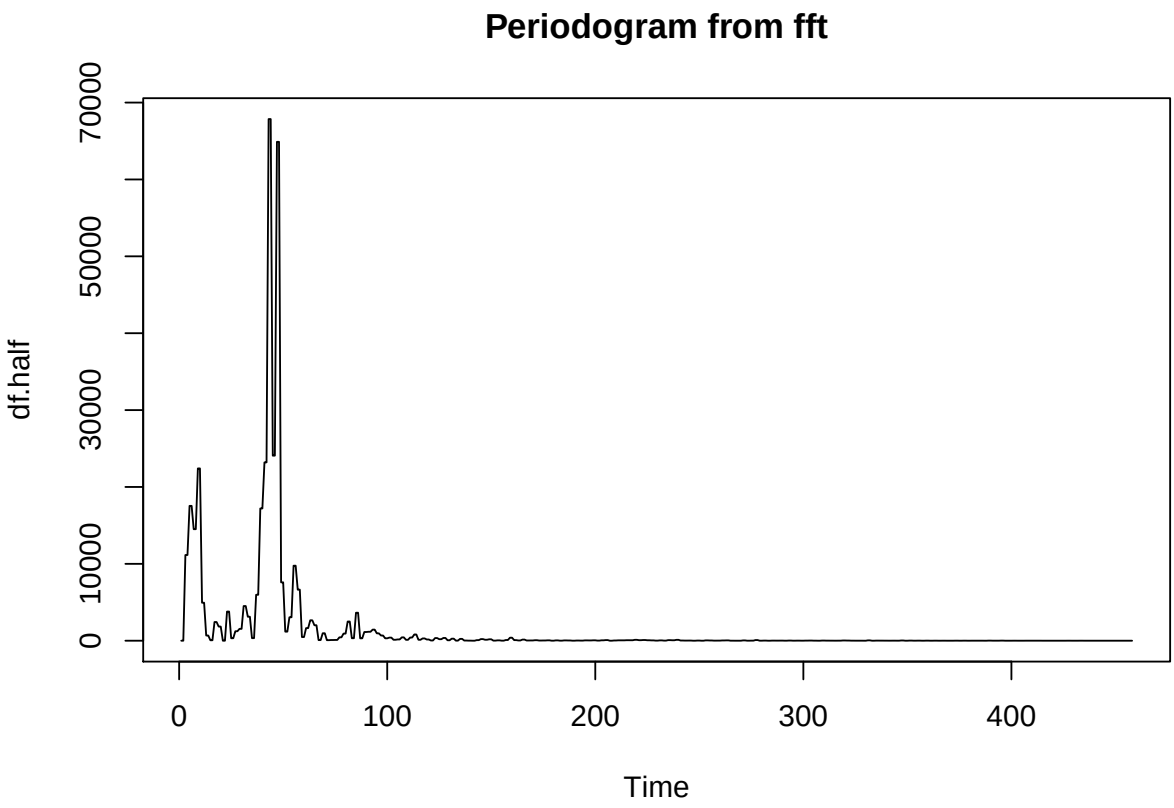
```
## [1] 749.1462
```

We see based on the Q-statistic that we have significant correlation structure remaining in the residuals. See comments below in section 3. We're not sure of this code.

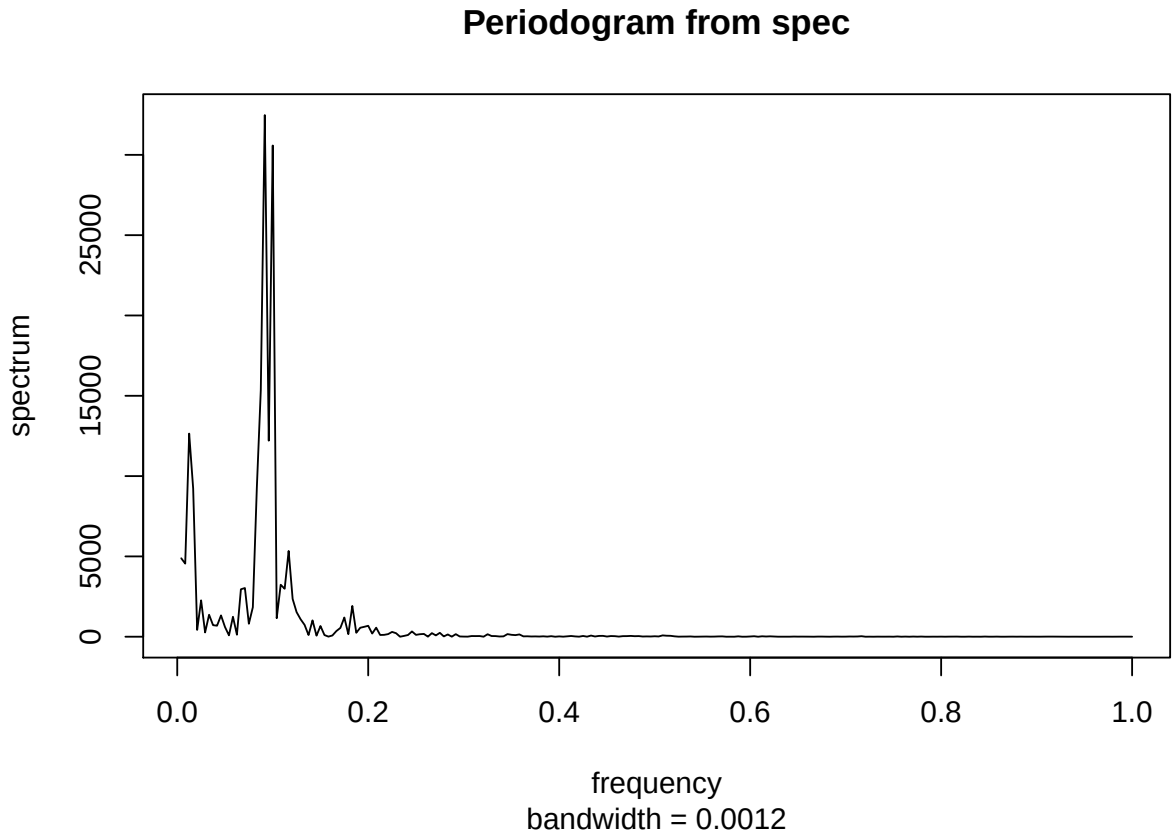
(c) Plot the periodogram of the sunspotz series using the spectrum command.

We used the definition of the periodogram and the fft to calculate $I(\omega_j) = |d(\omega_j)|^2$ taking into account that we had to normalize the fft result, take out the DC component, and only use half of the fft. We also plotted with the spec.pgram

```
df.periodogram <- Mod(fft(df - mean(df)))^2/length(df)
df.half <- ts(df.periodogram[1:length(df)/2])
plot(df.half, main = "Periodogram from fft")
```




```
df.periodogram.spec <- spec.pgram(df, taper = 0, log = "no", main = "Periodogram from sp
```



What we have plotted is the raw periodogram. The raw periodogram is not a consistent estimator for the spectral density. We could optionally use a smoothing kernel in the spec.pgram function to obtain a consistent estimate.

(d) Find the maximum of the periodogram values and the frequency at which the

maximum occurs.

We're interested in the max and the other peaks to try and understand the frequency components in the signal. Here we extract the max. The spec.pgram method returns a spectrum object with the data in `freq` , `spec` lists.

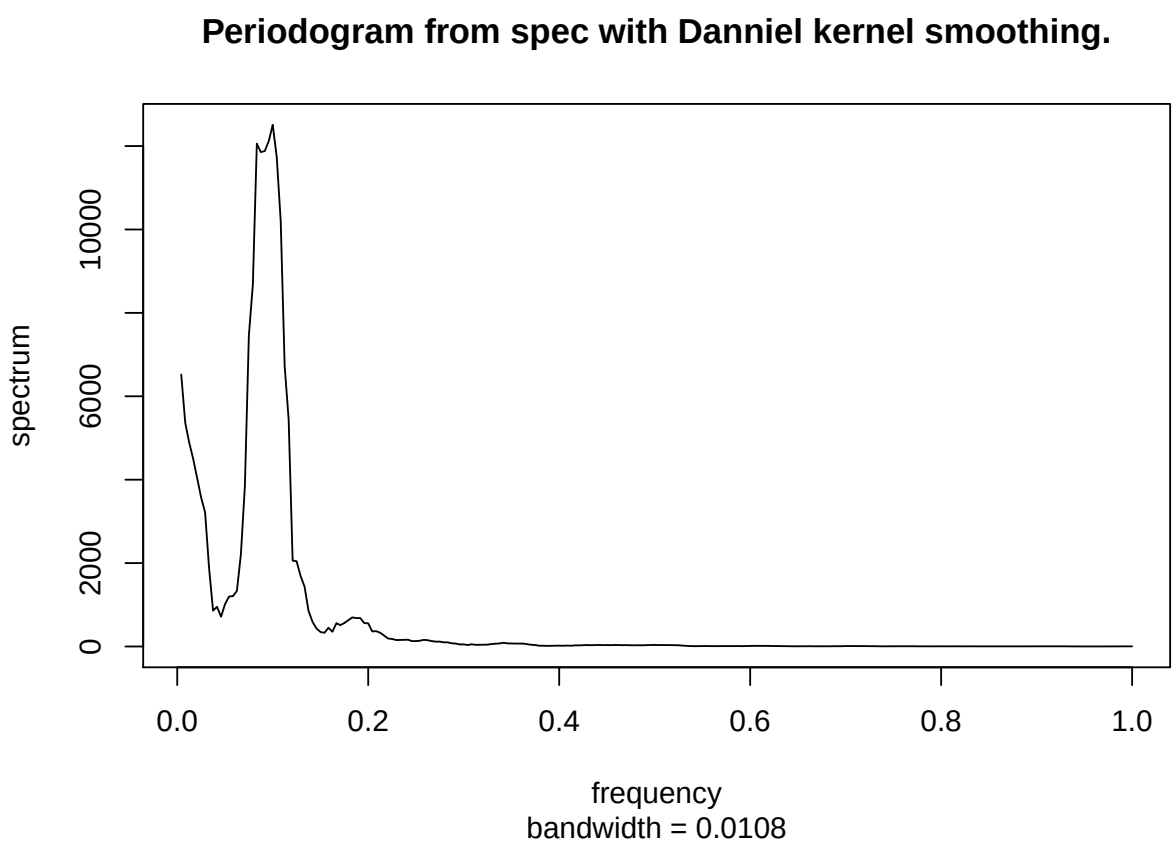
```
max.spec.loc <- which.max(df.periodogram.spec$spec)
max.spec <- df.periodogram.spec$spec[max.spec.loc]
max.spec.freq <- df.periodogram.spec$freq[max.spec.loc]

pander(data.frame(max.spec.freq = max.spec.freq, max.spec = max.spec))
```

max.spec.freq	max.spec
0.09167	32478

We tried to fix up some code to find locations of the other peaks in the periodogram. The idea is to use overlapping intervals to find the local peaks via a spline maximization. It doesn't work yet :(

We did try to smooth the spectrum with a Danniel Kernel first. We'll keep that plot for reference. Note the 2 periodic components are highlighted better in the Danniel smoothed spectrum.

```
k = kernel("daniell", 4)
df.periodogram.spec.danniel <- spec.pgram(df, k, taper = 0, log = "no", main = "Periodog
```



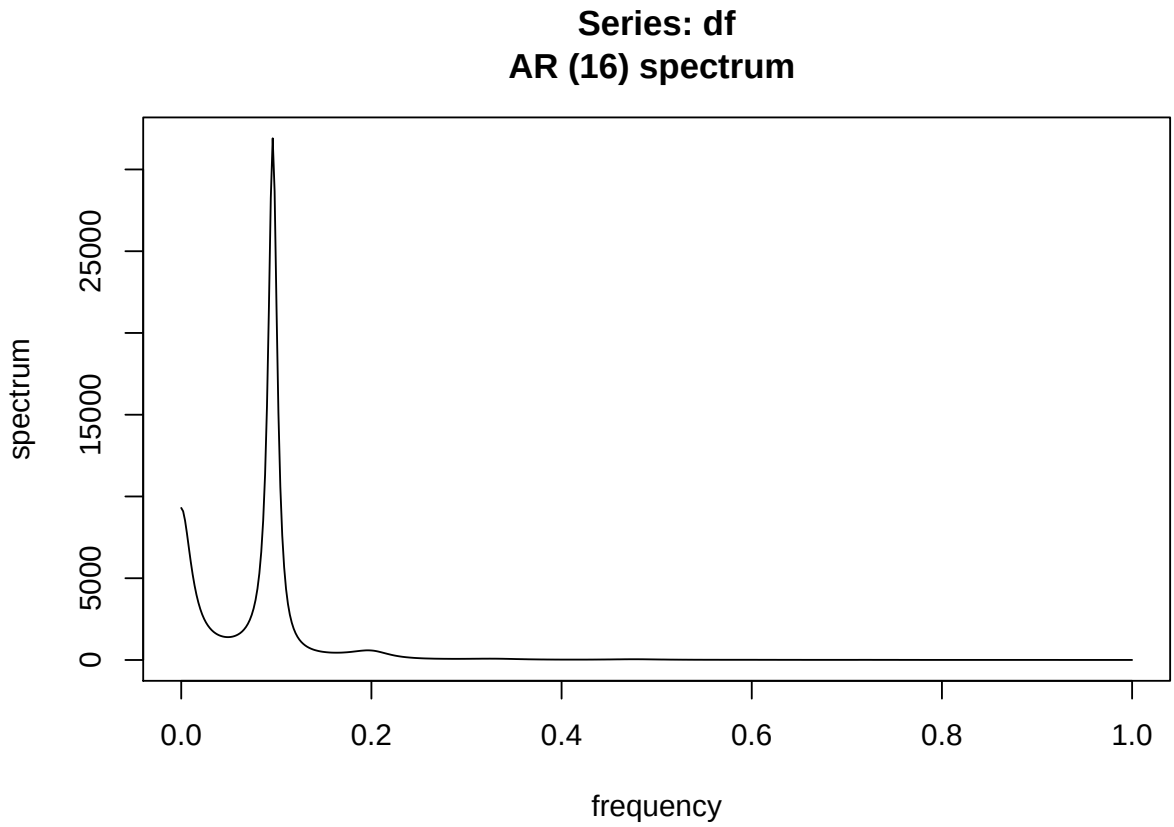

```
# f <- function(x, q, d) spline(q, d, xout = x)$y x <-
# df.periodogram.spec.danniel$freq y <- df.periodogram.spec.danniel$spec nb
# <- 20 # choose number of intervals iv <- embed(seq(floor(min(x)),
# ceiling(max(x)), len = nb), 2)[,c(2,1)] # make overlapping intervals to
# avoid problems if the peak is close to # the ends of the intervals (two
# modes could be found in each interval) iv[-1,1] <- iv[-nrow(iv),2] - 2 #
# The function 'f' is maximized at each of these intervals iv # the solution
# is a local maximum gr.thr <- 0.01 hes.thr <- 0.03i require('numDeriv')
# vals <- matrix(nrow = nrow(iv), ncol = 3) grd <- hes <- rep(NA,
# nrow(vals)) for (j in seq(1, nrow(iv))) { opt <- optimize(f = f, maximum =
# TRUE, interval = iv[j,], q = x, d = y) vals[j,1] <- opt$max vals[j,3] <-
```

```
# exp(opt$obj) grd[j] <- grad(func = f, x = vals[j,1], q = x, d = y) hes[j]
# <- hessian(func = f, x = vals[j,1], q = x, d = y) if (abs(grd[j]) < gr.thr
# && abs(hes[j]) > hes.thr) vals[j,2] <- 1 } # bin the peaks to avoid
# similar local maxima vals[,1] <- round(vals[,1], 2) if (anyNA(vals[,2])) {
# peaks <- unique(vals[-which(is.na(vals[,2])),1]) } else peaks <-
# unique(vals[,1]) plot(df.periodogram.spec$freq, df.periodogram.spec$spec,
# log = 'y', type = 'l') abline(v = peaks, lty = 2)
```

(e) Find an AR-based estimate of the spectral density of sunspotz using the command spec.ar.

First we note the caution from the documentation of spec.ar *Warning Some authors, for example Thomson (1990), warn strongly that AR spectra can be misleading.* We'll keep this in mind as we proceed.

```
df.spectrum.ar <- spec.ar(df, log = "no")  #, log='no', ylim=c(0,.5), plot = TRUE)
```



This looks similar to the Danniel smoothed spectrum only smoother. The second peak seems less distinct in this spectra.

(f) Find the maximum of the estimated spectral density and the frequency at which

the maximum occurs.

```
max.spec.loc <- which.max(df.spectrum.ar$spec)
max.spec <- df.spectrum.ar$spec[max.spec.loc]
max.spec.freq <- df.spectrum.ar$freq[max.spec.loc]

pander(data.frame(max.spec.freq = max.spec.freq, max.spec = max.spec))
```

max.spec.freq	max.spec
0.09619	31906

(g) Find the two estimates of the period of the most prominent periodic component

of the sunspotz series using your answers from parts (d) and (f). What does it say about the Sunspot cycle?

I'm not sure what we're being asked in this part. The period is given by $\frac{1}{\omega}$ so the max of the ar estimated spectral density has a period of $\frac{1}{0.09619}$ `10.3960911`

The max of the spectrum given by the fft is $\frac{1}{0.09167}$ `10.9086942` So the period is likely between 10-11 years, but it's not clear right now how we would statistically estimate this.

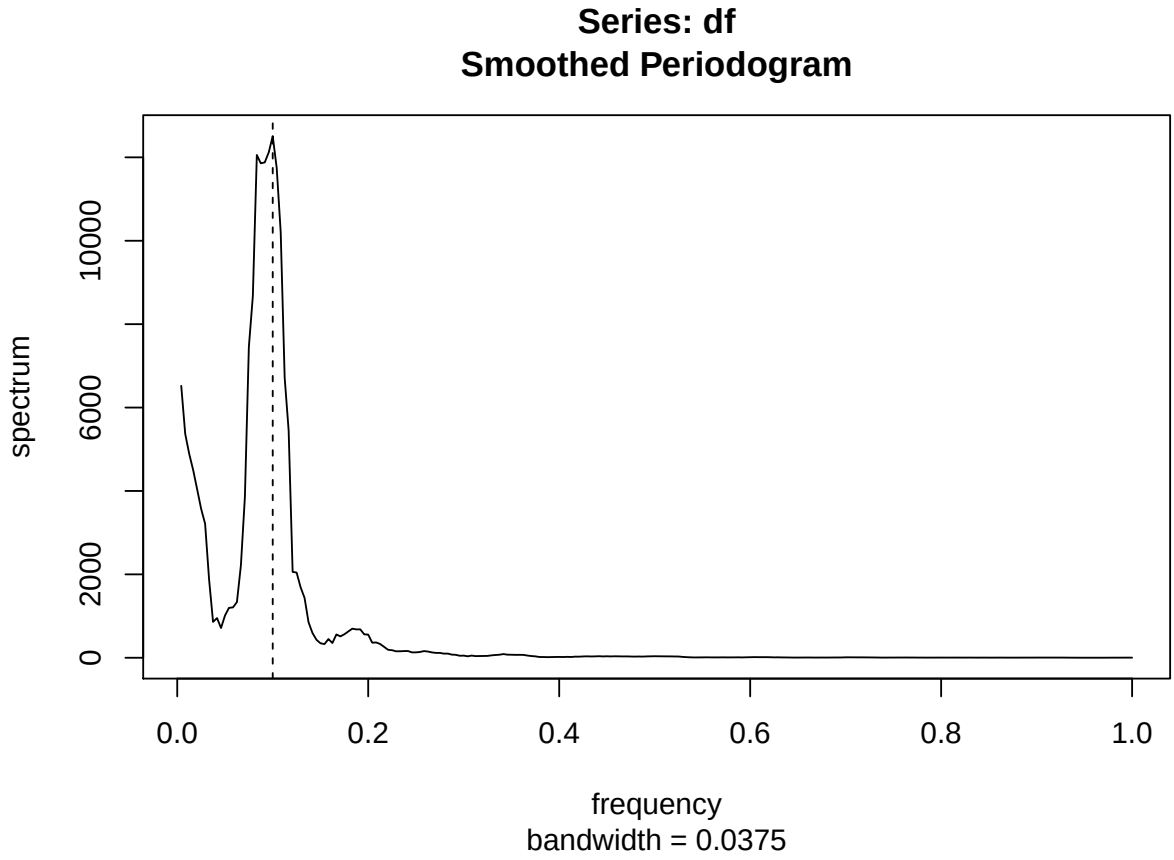
We can look at the frequency of the max of the Danniel smoothed spectrum for additional insight.

```
max.spec.loc <- which.max(df.periodogram.spec.danniel$spec)
max.spec <- df.periodogram.spec.danniel$spec[max.spec.loc]
max.spec.freq <- df.periodogram.spec.danniel$freq[max.spec.loc]
pander(data.frame(max.spec.freq = max.spec.freq, max.spec = max.spec))
```

max.spec.freq	max.spec
0.1	12511

We learned how to parametrically and non-parametrically estimate the spectrum $f(\omega_0)$ at a location ω_0 based on a window about ω_0. For fun let's calculate the 95%CI for $f(\omega)$ at the max of the Danniel smoothed series.

```
df.danniel.ave = mvspec(df, kernel("daniell", 4), log = "no")
abline(v = c(0.1), lty = 2)
```




```
max.spec.loc <- which.max(df.danniel.ave$spec)
df.danniel.ave$bandwidth
```

```
## [1] 0.0375
```

```
degree.freedom = df.danniel.ave$df
U = qchisq(0.025, degree.freedom)
L = qchisq(0.975, degree.freedom)
df.danniel.ave$spec[max.spec.loc]
```

```
## [1] 12510.58
```

```
# intervals
lower.ci <- degree.freedom * df.danniel.ave$spec[max.spec.loc]/L
upper.ci <- degree.freedom * df.danniel.ave$spec[max.spec.loc]/U

pander(data.frame(lower.ci = lower.ci, upper.ci = upper.ci), caption = "95 percent CI fo
```

lower.ci	upper.ci

Table 4: 95 percent CI for spectrum at max for Danniel smmothed. L=9

lower.ci	upper.ci
7066	27948

2 The spectrum of a linear combinatino of SOS processes

Suppose that $\{S_t\}$ and $\{N_t\}$ are zero mean SOS and independent time series with ACvFs $\gamma_1()$, γ_2 and and spectral densities $f_1(\omega)$ and $f_2(\omega)$, respectively. Let

$$X_t = S_t + AS_{t-D} + N_t$$

where $A \in (0, \infty)$ is a constant and $D > 0$ is an integer.

(a) Find the ACvF of $\{X_t\}$.

$$f_x(\omega) = [1 + A^2 + 2A\ cos(2\omega D)]f_1(\omega) + f2(\omega)$$

Without loss of generality we can assume that $E[S_t] = 0$ and $E[N_t] = 0$ otherwise we'll subtract μ_S and μ_N from S_t and N_t and $\mu_S + A\mu_S + \mu_N$ from X_t. Now that we've mean corrected, we can write all of the ACvF functions in the reduced form.

$$\gamma_X = E[X_{t+h}X_t] = E[(S_{t+h} + AS_{t+h-D} + N_{t+h})(S_t + AS_{t-D} + N_t)]$$

Since the processes are independent the cross covariance terms drop out and we're left with

$$\gamma_X = E[X_{t+h}X_t] = E[S_{t+h}S_t]+A^2\ E[S_{t-D+h}S_{t-D}]+E[N_{t+h}N_t]+A\ E[S_{t+h}S_{t-D}]+A\ E[S_{t-D+h}S_t]$$

Now we write this in terms of γ_S and γ_N

$$\gamma_X(h) = \gamma_S(h) + A^2\ \gamma_S(h) + \gamma_N(h) + A\ \gamma_S(h + D) + A\ \gamma_S(h - D)$$

(b) Using part (a) or otherwise, show that the spectral density of fXtg is given by

Looking good so far, we can see where it's going. But we need to make an assumption regarding the summability of γ_S and γ_N. If we have absolute summability of the autocovariance functions we'll be able to express the spectral densities in terms of the values of $\gamma(h)$. We'll make this assumption and move on. The plan is that we'll express $f_X(\omega)$ in terms of the densities of S and N and invoke the uniqueness of the Fourier transform.

$$f_X(\omega) = \sum_{h=-\infty}^{h=\infty} \gamma_X(h)e^{-2\pi i\omega h} = \sum_{h=-\infty}^{h=\infty} [\gamma_S(h)+A^2\ \gamma_S(h)+\gamma_N(h)+A\ \gamma_S(h+D)+A\ \gamma_S(h-D)\ ]\ e^{-2\pi i\omega h}$$

Now we note that

$$A \sum_{h=-\infty}^{h=\infty} \gamma_S(h+D)e^{-2\pi i\omega h} = A\ e^{2\pi i\omega D} \sum_{h=-\infty}^{h=\infty} \gamma_S(h+D)e^{-2\pi i\omega (h+D)} = A\ e^{2\pi i\omega D} \sum_{(h+D)=-\infty}^{(h+D)=\infty} \gamma_S(h+D)e^{-2\pi i\omega (h+D)}$$

$$= A\ e^{2\pi i\omega D} \sum_{h=-\infty}^{h=\infty} \gamma_S(h) e^{-2\pi i\omega(h)}$$

We get a similar result for the term with $h - D$ except for a sign change. Substituting these expressions and those for $f_S(\omega)$ and $f_N(\omega)$ into our equation for $f_X(\omega)$ we have that

$$f_X(\omega) = f_S(\omega) + A^2 f_S(\omega) + f_N(\omega) + A\ e^{2\pi i\omega D} f_S(\omega) + A\ e^{-2\pi i\omega D} f_S(\omega)$$

Collecting terms and using a trig identity, we have that.

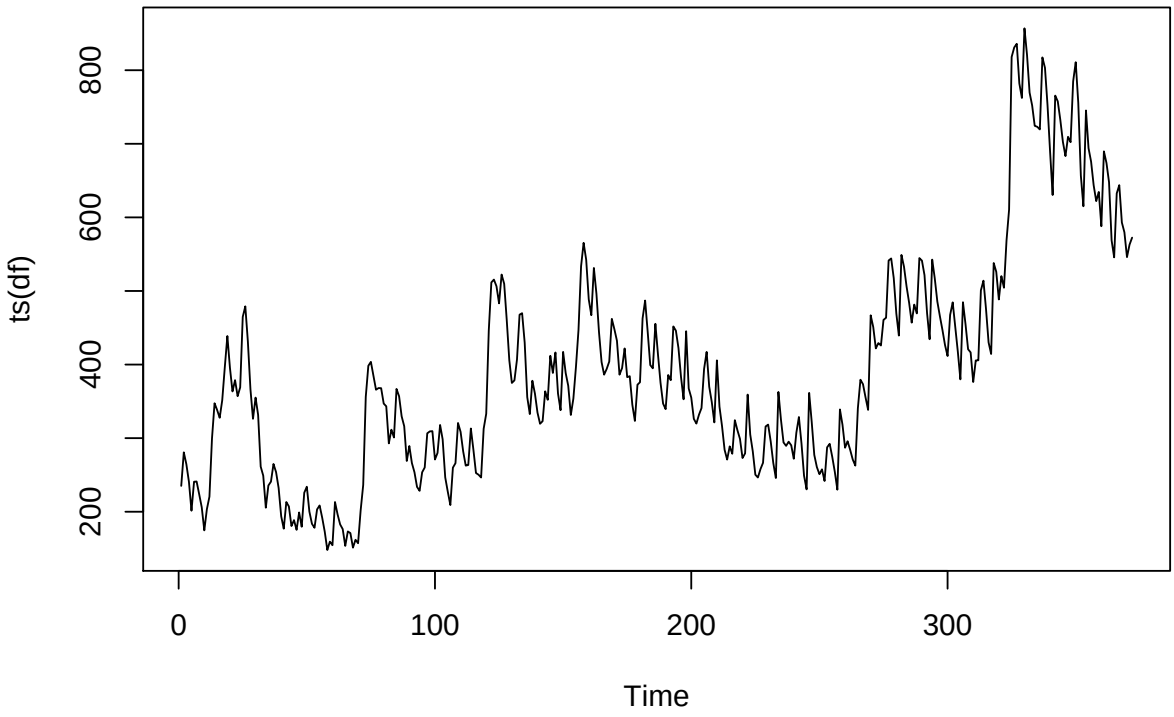
$$f_X(\omega) = [1 + A^2 + A\ cos(2\pi i\omega D)]\ f_S(\omega)\ + f_N(\omega)$$

Uniqueness of the Fourier transform assures us that the expression on the RHS is indeed the spectral density of X.

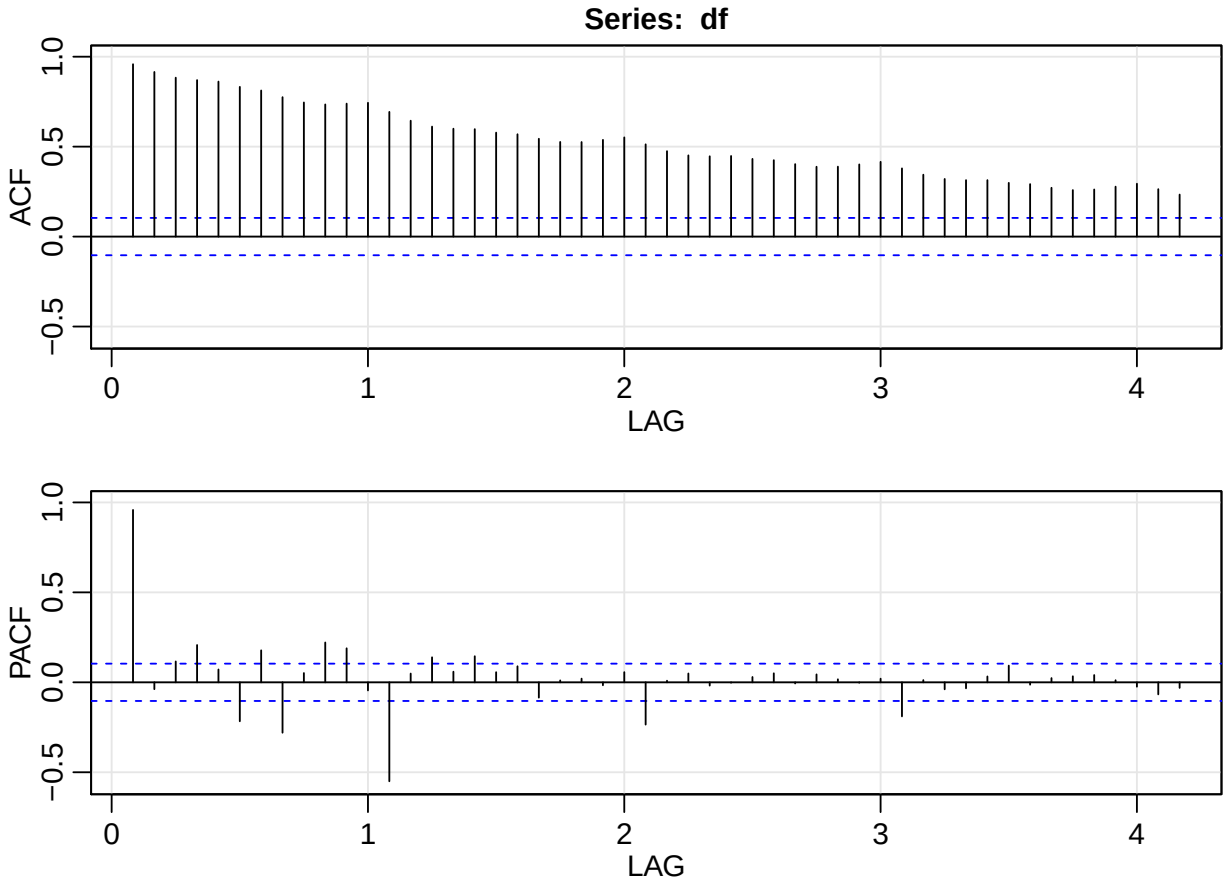
3. Consider the unemployment time series unemp given in the package astsa.

(a) Plot the unemp series and its ACF upto lag 50.

```
rm(list = ls())
library(astsa)
data(unemp, package = "astsa")
df <- unemp
plot(ts(df))
```

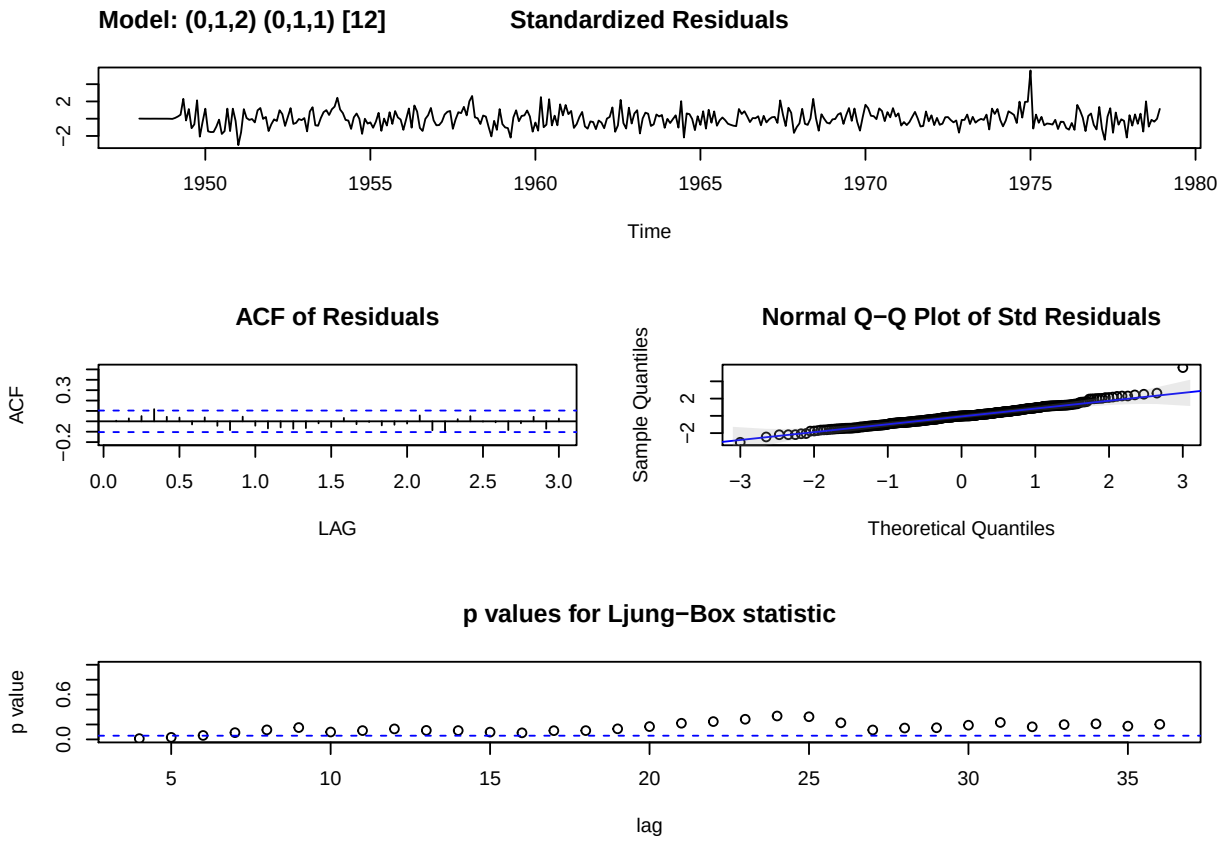



```
invisible(acf2(df, 50))
```



(b) Fit a Seasonal ARIMA model of order $(012) \times (0,1,1)_{12}$ and present the R output.

```
invisible(model <- sarima(df, 0, 1, 2, 0, 1, 1, 12))
```

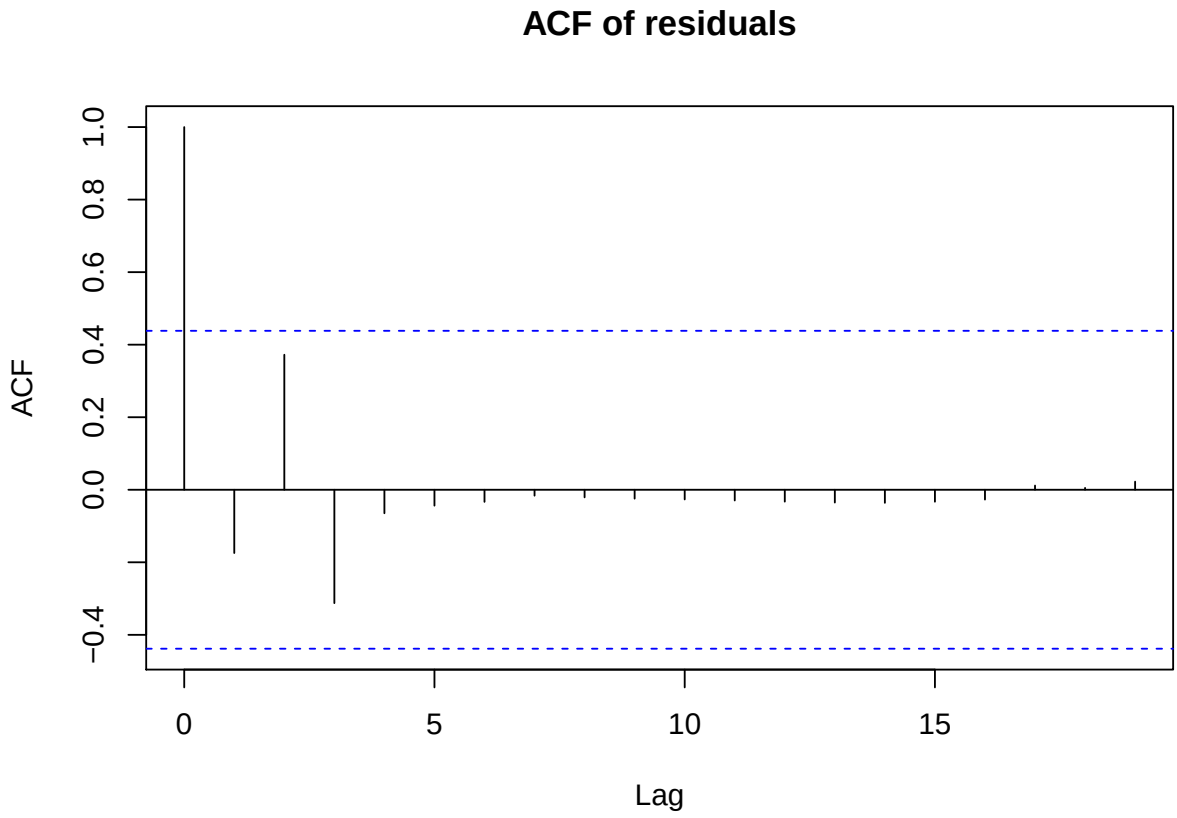



```
model$ttable
```

For lag 1 the Ljung-Box statistic shows significant correlation in the residuals.

Let's calculate the Ljung-Box-Pierce Q-statistic to check for systemic autocorrelation in the residuals. We'll extract the residuals and do the calculation by hand - there's and R function for this `Box-Test` that we've experimented with. we'll revisit this.

```
n <- length(df)
H <- 20
r <- model$fit$residuals[1:H]
acf.residuals <- acf(r, H, main = "ACF of residuals")
```




```
sum.denominator <- n - seq(H, 1, by = -1)
r.s <- acf.residuals$acf^2/sum.denominator
Q <- n * (n + 2) * sum(r.s)
Q
```

```
## [1] 506.2771
```

(c) Consider the Ljung-Box statistic p-value plot and the ACF of the residuals. What

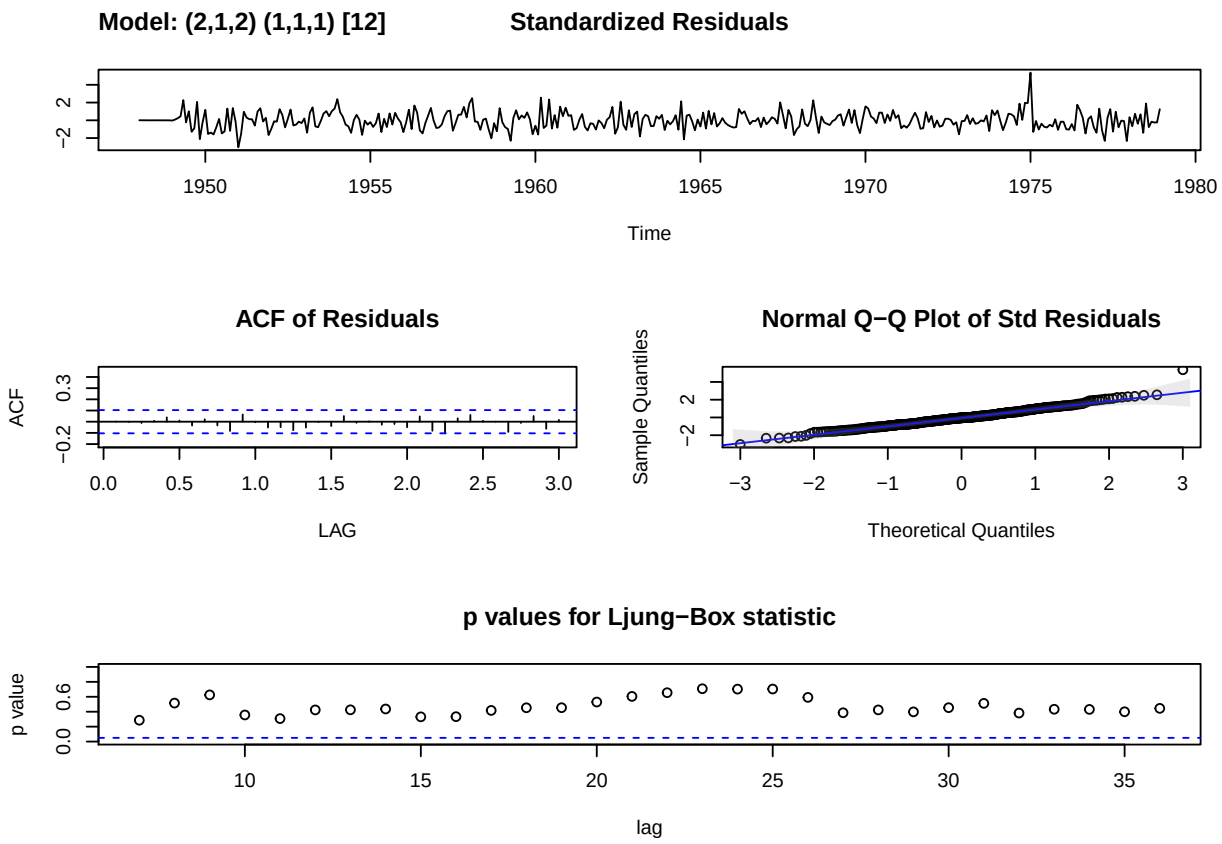
do these indicate about inadequacy of the fitted model ?

The Ljung-Box statistic shows nearly significant and autocorrelation in the residuals at some lags. We would look into increasing the AR order for our model to reduce this.

(d) Refine the SARIMA model based on your answer to the last part and present the

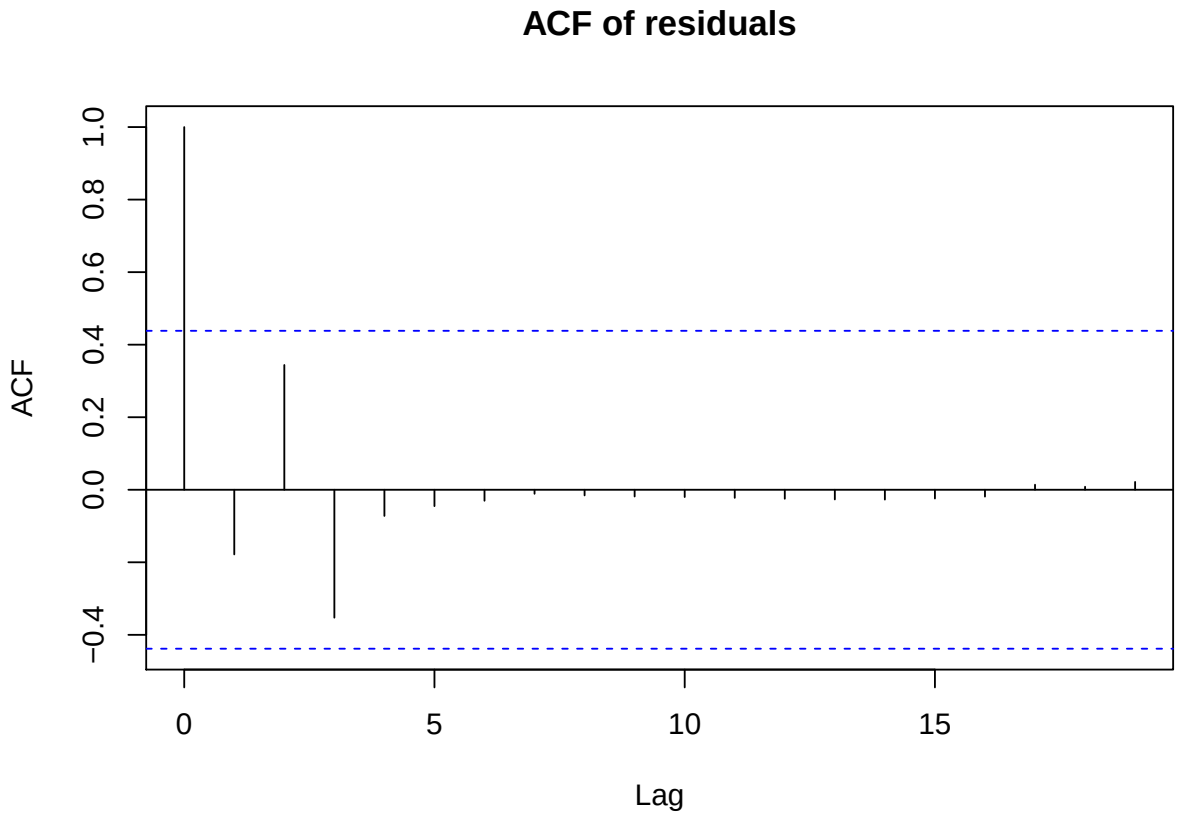
R output for the new SARIMA model. Indicate how the new model improves on the model in part (b).

```r
invisible(model <- sarima(df, 2, 1, 2, 1, 1, 1, 12))
```




```r
model$ttable
```

```r
n <- length(df)
H <- 20
r <- model$fit$residuals[1:H]
acf.residuals <- acf(r, H, main = "ACF of residuals")
```



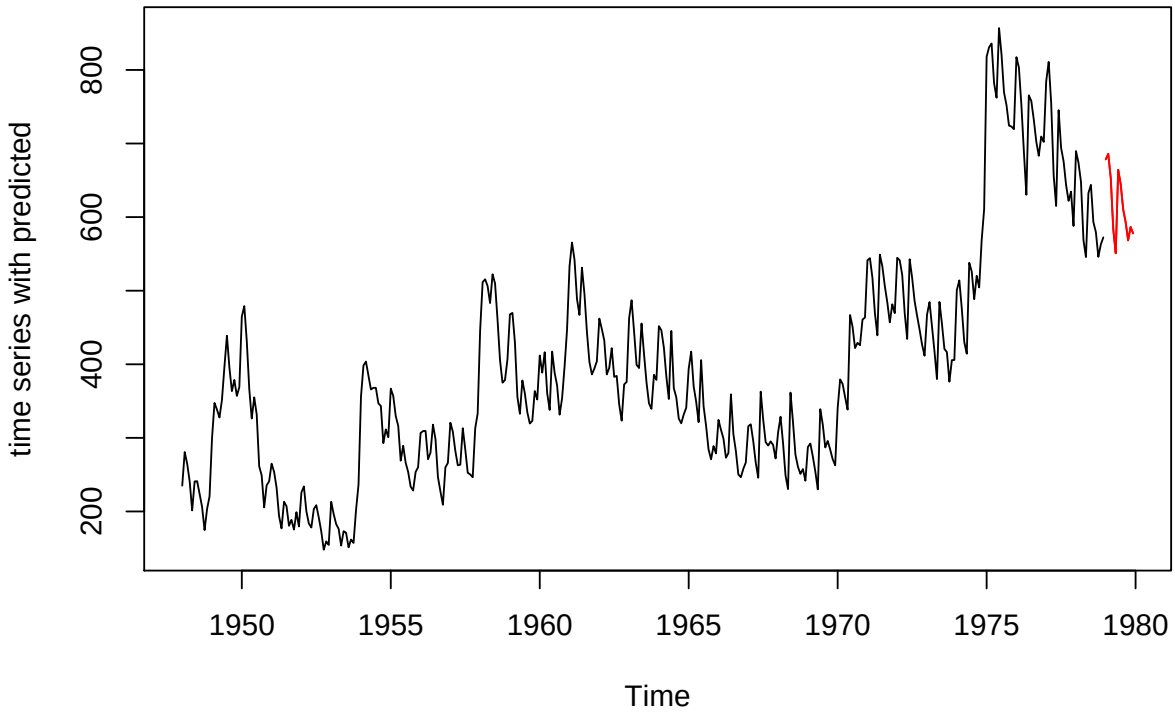

```
sum.denominator <- n - seq(H, 1, by = -1)
r.s <- acf.residuals$acf^2/sum.denominator
Q <- n * (n + 2) * sum(r.s)
Q
```

```
## [1] 508.2607
```

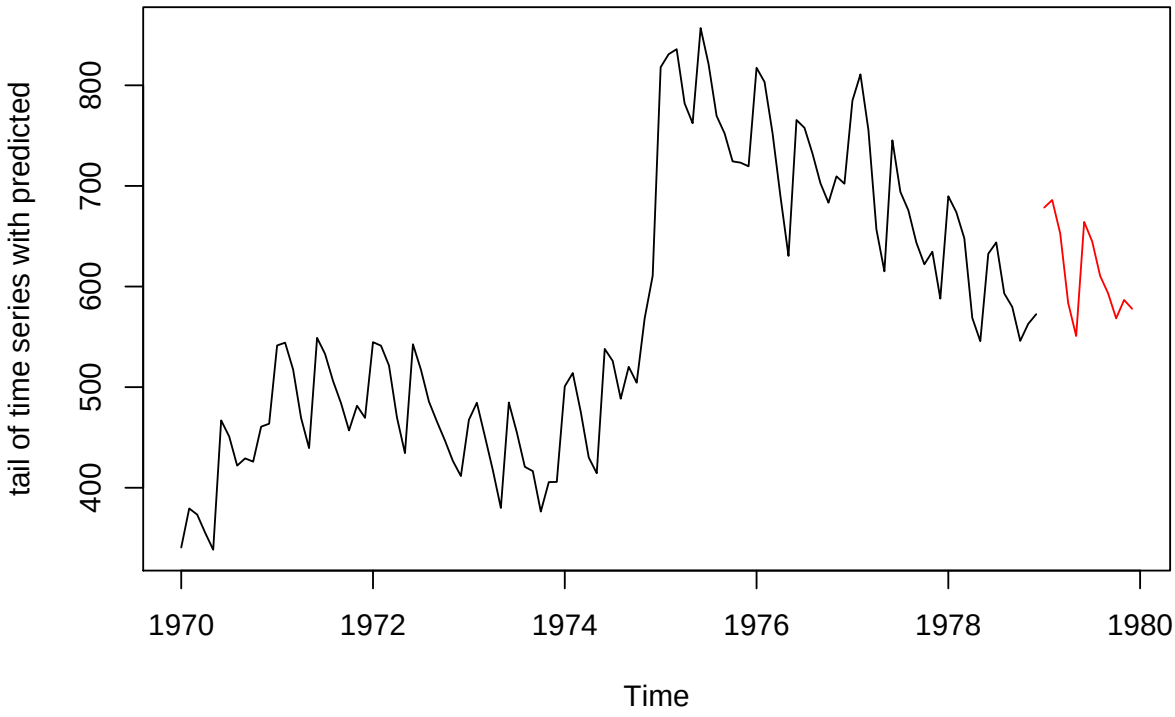
In looking at my calculation of the Ljung-Box-Pierce Q-statistic I'm suspicious of my code based on the plot above. REVISIT

(e) Use the fitted model in part (d) to forecast the next 12-months.

```
fore = predict(model$fit, n.ahead = 12, interval = "prediction")
ts.plot(df, fore$pred, col = 1:2, ylab = "time series with predicted")
lines(fore$pred, pch = "*", col = 2)
```




```
last <- window(df, start = 1970)
ts.plot(last, fore$pred, col = 1:2, ylab = "tail of time series with predicted")
```




```
fore$pred
```

```
##           Jan      Feb      Mar      Apr      May      Jun      Jul
## 1979 678.4418 685.9842 652.8264 583.7548 550.8145 664.1933 644.8671
##           Aug      Sep      Oct      Nov      Dec
## 1979 610.5690 593.0959 568.3710 586.6597 577.8968
```