Assignment #3 STA457H1F/2202H1F

Due Friday, November 15, 2024

Instructions: Solutions to problems 1 and 2 are to be submitted on Quercus (PDF files only). You are strongly encouraged to do problems 3 and 4 but these are **not** to be submitted for grading.

1. Data on the number of monthly traffic fatalities in Ontario from 1960 to 1974 are contained in a file `fatalities.txt` on Quercus. You may want to analyze the logs of the data.
(a) Use the function `stl` to seasonally adjust the data. (Some details on `stl` are available using the `help` facility in R – `help(stl)` – and more in the paper by Cleveland *et al.* on Quercus.
The two key tuning parameters in `stl` are `s.window` and `t.window`, which control the number of observations used by loess in the estimation of the seasonal and trend components, respectively; these parameters must be odd numbers. For example,

```
> fatal <- scan("fatalities.txt")
> fatal <- ts(fatal,start=c(1960,1),end=c(1974,12),freq=12)
> r <- stl(fatal,s.window = 3, t.window = 51, robust=T)
> plot(r)
> r <- stl(fatal,s.window = 5, t.window = 61, robust = T)
> plot(r)
> r <- stl(fatal,s.window = "periodic", t.window = 41, robust = T) # seasonal periodic
> plot(r)
```

The option `robust = T` allows one to better see anomalous observations or outliers in the irregular component.

(b) For one of set of parameter values used in part (a), look at the estimated irregular component. Does it look like white noise? Would you expect it to look like white noise?

(c) The function `stl` estimates trend, seasonal, and irregular components. Other seasonal adjustment procedures can also estimate a calendar component in order to reflect variation due to the number of weekend days, holidays etc. For these data, do you think a calendar component would be useful?

2. The file `speech.txt` contains a "speech record" of a person saying the syllable *ahh*; this was sampled at 10000 points per second. (These data represent a subset of a larger data set.) The data can be read into R as follows:

```
> speech <- ts(scan("speech.txt"),frequency=10000)
```

The argument `frequency=10000` reflects that fact that we have 10000 measurements per second and we are using seconds as the unit of measurement and Hertz (cycles per second) as the unit of frequency.

(a) Autoregressive spectral density estimates can be computed using `spec.ar`. For example,

```
> r <- spec.ar(speech,order=10,method="burg")
```

will give an estimate obtained by fitting an AR(10) model to the time series (using Burg's estimates) while

```
> r <- spec.ar(speech,method="burg")
```

will use AIC to choose the AR order (using Yule-Walker estimates). Again play around with different AR orders and compare them to the estimates in parts (b) and (c) below.

(b) Estimate the spectral density function using the multitaper method; the function `spec.mtm` is available in the package `multitaper`. The library can be loaded as follows:

```
> library(multitaper)
```

The two key parameters in `spec.mtm` are `nw`, the time-bandwidth parameter, and `k`, the number of tapers used to construct the estimate; `k` should be less than $2 \times$ `nw`.
Try different values of `nw` (and corresponding `k`).

(c) An R function `spec.parzen` is available for doing spectral density estimation using Parzen's lag window. For example,

```
> speech <- ts(scan("speech.txt"),frequency=10000)
> r <- spec.parzen(speech,lag.max=60,plot=T)
```

will compute the estimate using $M = 60$ (`lag.max=60`); the plot will give approximate pointwise 95% confidence intervals for the spectral density function. Play around with different values of M to see how the estimates change with M.

(d) Which frequencies (measured in Hertz or cycles per second) seem to be most dominant?

3. Suppose that $\{X_t\}$ is an ARIMA(p, d, q) process and define $\widehat{X}_{n+s}$ to be the best linear predictor of X_{n+s} given $X_n = x_n, X_{n-1} = x_{n-1}, \cdots$ Then

$$\sigma^2(s) = \sigma^2 \sum_{u=0}^{s-1} \psi_u^2$$

where σ^2 is the white noise variance and $\{\psi_u\}$ are defined by the identity

$$\sum_{u=0}^{\infty} \psi_u z^u = \frac{1 + \sum_{u=1}^{q} \beta_u z^u}{(1-z)^d (1 - \sum_{u=1}^{p} \phi_u z^u}$$

(a) Suppose that $\{X_t\}$ is an ARIMA(1,1,1) process with $\sigma^2 = 1$, $\phi_1 = 0.95$ and $\beta_1 = 0.2$. Evaluate $\sigma^2(s)$ for $s = 1, \cdots, 10$. (You may want to write a simple program in R to do the computations.)

(b) Suppose that $\{X_t\}$ is an ARMA(p,q) process whose parameters satisfy the usual conditions. Show that $\sigma^2(s) \to \text{Var}(X_t)$ as $s \to \infty$. (Hint: Note that X_t can be written as

$$X_t = \mu + \sum_{u=0}^{\infty} \psi_u \varepsilon_{t-u}$$

where $\{\varepsilon_t\}$ is white noise.)

4. Let $\{X_t\}$ be an MA(1) process

$$X_t = \varepsilon_t + \beta \varepsilon_{t-1}$$

where $\{\varepsilon_t\}$ is a zero mean white noise process, and $|\beta| < 1$. Let $\widehat{X_t}$ be the best linear predictor of X_t based on $X_1, \cdots, X_{t-1}$.

(a) Show that

$$\widehat{X_2} = \frac{\beta}{1+\beta^2} X_1.$$

(b) Show that

$$\widehat{X}_{n+1} = \frac{\beta}{\theta_n}(X_n - \widehat{X}_n)$$

where $\theta_1 = 1 + \beta^2$ and $\theta_n = 1 + \beta^2 - \beta^2/\theta_{n-1}$. (Note: This is quite difficult; you may be able to use the Levinson algorithm although it may be easier to work directly from the definition of the best linear predictor using the fact that $\rho(s) = 0$ for $s \geq 2$.)

(c) Show that $\lim_{n\to\infty} \theta_n = 1$. (Hint: You can use the following "fixed point theorem": Let f be a continuous function on some interval interval $[a, b]$ with $a \leq f(x) \leq b$ for $a \leq x \leq b$. Define a sequence $\{\theta_n\}$ such that $\theta_n = f(\theta_{n-1})$. If $|f'(x)| \leq k < 1$ for $a < x < b$ and $a \leq \theta_i \leq b$ for some i then $\lim_{n\to\infty} \theta_n = \theta_0$ where $f(\theta_0) = \theta_0$.)