

## Chapter 11. Time Series Analysis

```
# Read in GX 5-1 data and create time series
```

```
GX.dat <- scan("http://astrostatistics.psu.edu/MSMA/datasets/GX.dat")
GX.time <- seq(from=0, to=512, length.out=65536)
GX.ts <- ts(GX.dat, GX.time) ; GX.ts.offset <- ts(GX.dat-30, GX.time)
```

```
# Compare histogram of counts to normal distribution
```

```
hist(GX.dat, breaks=100, xlim=c(40,100), ylim=c(0,3500), xlab='GX 5-1 counts',
     font=2, font.lab=2, main='')
curve(dnorm(x,mean=mean(GX.dat), sd=sqrt(mean(GX.dat)))*65536, lwd=3, add=T)
sd(GX.dat) / sqrt(mean(GX.dat)) # result is 1.24
```

```
# Examine raw and smoothed time series
```

```
plot.ts(GX.ts[1:6000], ylab='GX 5-1 counts', xlab='Time (1/128 sec)',
        cex.lab=1.3, cex.axis=1.3)
plot(GX.time,GX.dat, ylim=c(-10,115), xlab='Time (sec)', ylab="GX 5-1 counts",
     cex.lab=1.3, cex.axis=1.3, type='n')
lines(ksmooth(GX.time, GX.dat+30, 'normal', bandwidth=7), lwd=2)
text(450, 110, 'Normal kernel')
lines(filter(GX.ts, sides=2, rep(1,7)/7), lwd=2)
text(450, 85, 'Moving average')
lines(kernapply(GX.ts.offset, kernel('modified.daniell', 7)), lwd=2)
text(450, 50, 'Modified daniell')
lines(supsmu(GX.time, GX.dat-60), lwd=2)
text(400, 20, "Friedman's supersmoother")
lines(lowess(GX.time, GX.dat-80, 0.05), lwd=2)
text(450, 0, "Local regression")
```

```
# Raw periodogram
```

```
f <- 0:32768/65536
I <- (4/65536) * abs(fft(GX.ts) / sqrt(65536))^2
plot(f[2:60000], I[2:60000], type="l", ylab="Power", xlab="Frequency")
```

```
Pergram <- spec.pgram(GX.ts,log='no',main='')
summary(Pergram)
Pergram$spec[500] # value of 500th point
2*Pergram$spec[500] / qchisq(c(0.025,0.975),2)
```

```
# Raw and smoothed periodogram
```

```
par(mfrow=c(3,1))
```

```

spec.pgram(GX.ts, log='no', main='', sub='')
spec.pgram(GX.ts, spans=50, log='no', main='', sub='')
spec.pgram(GX.ts, spans=500, log='no', main='', sub='')

# Autocorrelation functions of the GX 5-1 time series

par(mfrow=c(1,2))
acf(GX.ts, 40, main='', ci.col='black', ylim=c(-0.05,0.3), lwd=2)
pacf(GX.ts, 40, main='', ci.col='black', ylim=c(-0.05,0.3), lwd=2)

# Autoregressive modeling

ARmod <- ar(GX.ts, method='ols')
ARmod$order # model selection based on AIC
ARmod$ar # best-fit parameter values
ARmod$asy.se.coef$ar # parameter standard errors
plot(0:29, log10(ARmod$aic[1:30]), xlab='AR model parameters',
     ylab='log(AIC)', pch=20)
arrows(27, 0.4, 27, 0.0, length=0.1)

# Spectrum of AR model

ARspec <- spec.ar(GX.ts, plot=F)
GXspec <- spec.pgram(GX.ts, span=101, main='', sub='', lwd=2)
lines(ARspec$freq, ARspec$spec, col='red', lwd=4)
legend(0.23, 550, c('Periodogram, Daniell smooth', 'AR(27) model'),
      lty=c(1,1), lwd=c(2,4), col=c('black','red'))

# Spectral estimates of the long-range memory parameter d

install.packages('fracdiff') ; library(fracdiff)
d.FARIMA <- fracdiff(GX.ts, nar=27, nma=1, ar=ARmod$ar) ; d.FARIMA$d
d.GPH <- fdGPH(GX.ts)
d.Reisen <- fdSperio(GX.ts)

# Time domain estimates of the long-range memory parameter H=d+1/2

install.packages('fractal') ; library(fractal)
d.DFA <- DFA(GX.ts) ; d.DFA
plot(d.DFA)
d.ACVF <- hurstACVF(GX.ts) ; d.ACVF
d.block <- hurstBlock(GX.ts) ; d.block

# Discrete wavelet transform

install.packages('waveslim') ; library(waveslim)

```

```
GX.wav <- dwt(GX.ts,n.levels=10)
par(mfrow=c(3,1))
plot.ts(up.sample(GX.wav[[4]],2^4),type='h',axes=F,ylab='') ; abline(h=0)
plot.ts(up.sample(GX.wav[[7]],2^7),type='h',axes=F,ylab='',lwd=2) ; abline(h=0)
plot.ts(up.sample(GX.wav[[10]],2^{10}),type='h',axes=F,ylab='',lwd=2) ; abline(h=0)
par(mfrow=c(1,1))
```