Time Series Analysis

Eric Feigelson (Penn State) edf@astro.psu.edu

2nd East Asian Workshops in Astrostatistics Summer 2018

Adapted from R scripts in *Modern Statistical Methods for Astronomy With R Applications*, Eric D. Feigelson & G. Jogesh Babu 2012 http://astrostatistics.psu.edu/MSMA (http://astrostatistics.psu.edu/MSMA)

The MSMA textbook has a variety of R procedures designed for evenly spaced time series. Here we present some of the few procedures appropriate for unevenly spaced time series.

We start with a sparse dataset of radial velocity measurements of a bright star. This is a case where the stellar motion can be attributed to a Neptune-size planet in an eccentric orbit.

In [ ]:
```
setwd('/Users/ericfeigelson/Desktop/Rdir')

###
### HD 3651 radial velocities: bserved and folded time series
###

rv <- read.table('HD3651_rv.dat')[1:3]
names(rv) <- c('Day', 'Vel', 'sigVel')

install.packages('Hmisc', repos='https://cloud.r-project.org')  ;  lib
rary(Hmisc)
plot(rv[,1], rv[,2], pch=20)
errbar(rv[,1], rv[,2], (rv[,2]+rv[,3]), (rv[,2]-rv[,3]), xlab='JD - 24
4000',
       ylab='Velocity (m/s)', cap=0.01)
text(10000, 30, 'HD 3651    P=62.257d   e=0.63', pos=4)
```

We now compute the Lomb-Scargle periodogram and confirm the published period of P=62.257 days. The next plot shows a local polynomial regression fitted to the data folded with this period.

In [ ]:
```
# Lomb-Scargle periodogram for HD 3651

install.packages('lomb', repos='https://cloud.r-project.org')  ;  library(lomb)
lsp(rv[,1:2], from=0.00, to=0.05, ylab='Lomb-Scargle', alpha=NULL, main='')
arrows(1/62.257, 35, 1/62.257, 30, lwd=2, length=0.1)

install.packages('locfit', repos='https://cloud.r-project.org')  ;  library(locfit)
locfit_phase <- locfit(rv[,2] ~ lp(rv[,1] %% 62.257, nn=0.5), weights=1/rv[,3]^2)
plot(locfit_phase, ylim=c(-30,20), band='local',
        xlab='JD - 2440000 mod P=62.257d', ylab='Velocity (m/s)')
points(rv[,1] %% 62.257, rv[,2], pch=20)
```

CRAN package `RobPer` implements several periodograms for irregularly spaced dadta: Schwarzenberg-Czerny's Analysis of Variance and Stellingwerf's phase dispersion minimization are arithmetically related, while the other two are based on spline fits.

In [ ]:
```
# Various astronomical periodograms for HD 3651

install.packages('RobPer', repos='https://cloud.r-project.org')  ; lib
rary(RobPer)
RobPer_per <- seq(from=1, to=1000, length.out=10000)
RobPer_freq <- 1/RobPer_per

par(mfrow=c(2,2))  ;  par(mar=c(5,4,1,1)) ; par(mgp=c(2,1,0))
HD3651_AOV <- RobPer(rv[,1:3], weighting=TRUE,  regression='L2',
        model='step', steps=10, periods=RobPer_per)
plot(RobPer_freq, HD3651_AOV, type='l', xlab='frequency', ylab='AOV',
        xlim=c(0,0.03), ylim=c(0,0.85))
arrows(1/62.257, 0.85, 1/62.257, 0.75, length=0.05)

HD3651_PDM <- RobPer(rv[,1:3], weighting=TRUE,  regression='L2',
        model='2step', steps=10, periods=RobPer_per)
plot(RobPer_freq, HD3651_PDM, type='l', xlab='frequency', ylab='PDM',
        xlim=c(0,0.03), ylim=c(0,0.80))
arrows(1/62.257, 0.80, 1/62.257, 0.70, length=0.05)

HD3651_L1 <- RobPer(rv[,1:3], weighting=TRUE,  regression='L1',
        model='sine', steps=10, periods=RobPer_per)
plot(RobPer_freq, HD3651_L1, type='l', xlab='frequency', ylab='L1',
        xlim=c(0,0.03), ylim=c(0,0.4))
arrows(1/62.257, 0.40, 1/62.257, 0.35, length=0.05)

HD3651_GCV <- RobPer(rv[,1:3], weighting=TRUE,  regression='L2',
        model='splines', steps=10, periods=RobPer_per)
plot(RobPer_freq, HD3651_GCV, type='l', xlab='frequency', ylab='GCV',
        xlim=c(0,0.03), ylim=c(0,0.70))
arrows(1/62.257, 0.70, 1/62.257, 0.63, length=0.05)
```

We now turn to two of ~200,000 light curves obtained during the prime mission of NASA's Kepler satellite. Here the purpose is to detect faint periodic dips caused by transits of orbiting planets. But the main difficulty is autocorrelated variability in the host star. The light curves are evenly spaced with observations every 24.9 minutes for ~4 years, but some time slots have missing data.

We start with plots of the light curves and proceed with nonparametric analysis of the first star: its autocorrelation function, and tests for normality and trend in the flux distribution.

In [ ]:
```
###
### Ingest and plot light curves for two Kepler stars
###

Kepler1 <- read.table('Kepler1.dat')[[1]]  # KIC 007596240
Kepler2 <- read.table('Kepler2.dat')[[1]]  # KIC 007609553

length(which(is.na(Kepler1))) / length(Kepler1)  # 16% NAs
length(which(is.na(Kepler2))) / length(Kepler2)  # 29% NAs

par(mfrow=c(2,1))  ;  par(mar=c(5,4,1,2))
plot(Kepler1, type='l', xlab='Time')
plot(Kepler2, type='l', xlab='Time')



###
### Properties of the Kepler 1 lightcurve (KIC 007596240)
###

median(Kepler1, na.rm=TRUE)
cat('InterQuartile Range(Kepler1) = ',
     IQR(Kepler1, na.rm=TRUE), '\n')

par(mfrow=c(3,1))  ;  par(mar=c(5,4,1,2))
acf(Kepler1, na.action=na.pass, ylim=c(-0.05, 0.2),
         xlab='Kepler 1 lag', main='',  ci.col='black')

hist(Kepler1, freq=FALSE, breaks=200, main='', xlim=c(-20,20),
        xlab='Kepler 1 values')
library(MASS)  # Comparison with normal distribution
Kepler1_noNA <- na.omit(Kepler1)
Kep1_mn <- fitdistr(Kepler1_noNA, 'normal')[[1]][1]
Kep1_sd <- fitdistr(Kepler1_noNA, 'normal')[[1]][2]
curve(dnorm(x, Kep1_mn, Kep1_sd), -20, 20, add=TRUE)

install.packages('nortest', repos='https://cloud.r-project.org')
library(nortest)  ;  ad.test(Kepler1)

cor.test(1:length(Kepler1), Kepler1, method='kendall')  # Tests for tr
end
```

We proceed with parametric autoregressive modeling of the quiet star Kepler1 with ARIMA, and perform nonparametric tests on the ARIMA residuals. We find that a low dimensional ARIMA(2,1,2) model does not reduce the IQR (variability amplitude) but does greatly reduce the autocorrelation.

In [ ]:
```
# Parametric modeling with ARIMA

Kep1_arima <- arima(Kepler1, order=c(2,1,2))  #  Autoregressive model
str(Kep1_arima)
cat('InterQuartile Range(ARIMA residuals for Kepler1) = ',
     IQR(Kep1_arima$residuals, na.rm=TRUE), '\n')
acf(Kep1_arima$residuals, na.action=na.pass, ylim=c(-0.05, 0.2),
        xlab='Kepler 1 lag', ci.col='black')
```

We then run time series diagnostics commonly used in econometrics: Box-Pierce test for independence, Augmented Dickey-Fuller test for stationarity, and BDS test for nonlinear serial dependence.

In [ ]:
```
install.packages('imputeTS', repos='https://cloud.r-project.org')
library(imputeTS)
arima_resids <- na.kalman(Kep1_arima$residuals)

Box.test(arima_resids)                   #  test for independence

install.packages('tseries', repos='https://cloud.r-project.org') ; lib
rary(tseries)
adf.test(arima_resids)                   # tests for stationarity (CPU
intensive)
bds.test(arima_resids)                   # test for i.i.d.
```

Turning to the Kepler2 star exhibiting rotational modulated starspots, we first impute (interpolate) missing values using an ARIMA interpolator. Note how the model based on the full light curve does an excellent job in estimating the nonlinear shape of variations in the missing intervals.

In [ ]:
```
###
### Properties  of Kepler 2 lightcurve (KIC 007609553)
###

# Imputation of missing values

Kepler2_impute <- na.kalman(Kepler2)   # impute NAs with Kalman smooth
er
par(mfrow=c(2,1))  ;  par(mar=c(5,4,1,2))
plot(Kepler2, type='l', xlim=c(60000,70000), xlab='Time', ylab='Kepler
2')
plot(Kepler2_impute, type='l', xlim=c(60000,70000), xlab='Time',
        ylab='Kepler 2 imputed')
```

We now show three different power spectra for Kepler2: a Fourier power spectrum based on the interpolated lightcurve; a Lomb-Scargle periodogram based on the original lightcurve: and an unweighted Stellingwerf phase dispersion minimization periodogram. The Lomb-Scargle periodigram shows the strongest periodicity with the reduced aliasing.

In [ ]:
```
# Three periodograms: Fourier, Lomb-Scargle, epoch folding

par(mfrow=c(3,1)) ;  par(mar=c(5,4,1,2))
spec.pgram(Kepler2_impute, xlim=c(0,0.005), spans=5, taper=0.0,
        main='', ylab='Fourier', sub='')

lsp(Kepler2, from=0.00, to=0.005, ylab='Lomb-Scargle', main='')

Kepler2_temp <- cbind(1:length(Kepler2), Kepler2)
Kepler2_irreg <- Kepler2_temp[!is.na(Kepler2_temp[,2]),]
PDM_per <- seq(from=1, to=10001, length.out=1000)
Kepler_PDM <- RobPer(Kepler2_irreg, weighting=FALSE,  regression='L2',
        model='step', steps=10, periods=PDM_per)        # (CPU intensi
ve)
PDM_freq <- 1/PDM_per
plot(PDM_freq, Kepler_PDM, type='l', xlab='frequency', ylab='PDM',
        xlim=c(0,0.005))
```

An alternative approach to harmonic analysis or ARIMA-type parametric modeling is wavelet analysis. We apply a discrete wavelet transform to the interpolated Kepler2 light curve and plot the wavelet coefficients at three scales. The longer scales show the quasi-periodic rotational modulation, while the shorter scales show rapid excursions.

In [ ]:
```
# Discrete wavelet transform for Kepler 2 (KIC 010191257)

#install.packages('waveslim', repos='https://cloud.r-project.org') ; l
ibrary(waveslim)
Kepler2_impute <- na.kalman(Kepler2)   # impute NAs with Kalman smooth
er
Kepler2_wavdat <- Kepler2_impute[1:2^16]
Kepler2_dwt <-  dwt(Kepler2_wavdat,n.levels=10)
par(mfrow=c(3,1))
plot.ts(up.sample(Kepler2_dwt[[5]],2^5), type='h', ylab='') ; abline(h
=0)
plot.ts(up.sample(Kepler2_dwt[[7]],2^7), type='h', ylab='', lwd=2) ;
abline(h=0)
plot.ts(up.sample(Kepler2_dwt[[9]],2^{9}), type='h', ylab='', lwd=2) ;
abline(h=0)
```

As a final step, we apply wavelet denoising for the Kepler2 light curve, setting small wavelet coefficients to zero and applying an inverse wavelet transform back to the time domain. The result is a superb improvement in the heteroscedastic noise superposed on the rotational modulations.

In [ ]:
```
# Wavelet denoising

install.packages('wavethresh', repos='https://cloud.r-project.org')  ;
library(wavethresh)
Kepler2_wd <- wd(Kepler2_wavdat)
Kepler2_wdth <- threshold(Kepler2_wd, policy='universal')
Kepler2_thresh <- wr(Kepler2_wdth)
par(mfrow=c(2,1))  ;  par(mar=c(4,4,1,1))
plot(Kepler2_wavdat, type='l', xlim=c(30000,40000), ylim=c(-200,200),
ylab='Kepler 2')
plot(Kepler2_thresh, type='l', xlim=c(30000,40000), ylim=c(-200,200),
ylab='Kepler 2')
```