Time Series Analysis (ARIMA)

Nik Bear Brown

In this lesson we'll learn the how to implement Time series analysis using ARIMA in R.

Additional packages needed

To run the code you may need additional packages.

- If necessary install the followings packages.

```
install.packages("RCurl");
install.packages("plyr");
install.packages("forecast");
```

```
library(RCurl)

## Loading required package: bitops

library(plyr)
library(forecast)

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: timeDate

## This is forecast 7.3
```

Data

We will be using is some Seismic Activity in Greece (earthquake) data from the University of Athens, Greece http://www.geophysics.geol.uoa.gr/

```
data <- read.table(textConnection(getURL(
  "http://www.geophysics.geol.uoa.gr/catalog/catgr_20002008.epi")),sep
= "", header = F)
#  The last two columns of data contain the surface depth and
earthquake intensity.
names(data) <- c("date", "mo", "day", "hr", "mn", "sec",
```

```
                              "lat", "long", "depth", "mw")
head(data, n = 3)

##   date mo day hr mn  sec     lat  long depth  mw
## 1 2000  1   1  1 19 28.3 41.950N 20.63     5 4.8
## 2 2000  1   1  4  2 28.4 35.540N 22.76    22 3.7
## 3 2000  1   2 10 44 10.9 35.850N 27.61     3 3.7
```

Autoregressive integrated moving average (ARIMA)

An autoregressive integrated moving average (ARIMA or ARMA) model combines an autoregressive component with a moving average component in to a single model.

An autoregressive integrated moving average (ARIMA or ARMA) model is a generalization of an autoregressive moving average (ARMA) model. These models are fitted to time series data either to better understand the data or to predict future points in the series (forecasting). They are applied in some cases where data show evidence of non-stationarity, where an initial differencing step (corresponding to the "integrated" part of the model) can be applied to reduce the non-stationarity.

Non-seasonal ARIMA models are generally denoted $ARIMA(p, d, q)$ where parameters $p, d, andq$ are non-negative integers, p is the order of the Autoregressive model, d is the degree of differencing, and q is the order of the Moving-average model. The he number of differences d is determined using repeated statistical tests. The values of p and q are then chosen by minimizing the AICc after differencing the data d times.

The ARIMA model uses an iterative three-stage modeling approach:

Model identification and model selection: making sure that the variables are stationary, identifying seasonality in the dependent series (seasonally differencing it if necessary), and using plots of the autocorrelation and partial autocorrelation functions of the dependent time series to decide which (if any) autoregressive or moving average component should be used in the model.

Parameter estimation using computation algorithms to arrive at coefficients that best fit the selected ARIMA model. The most common methods use maximum likelihood estimation or non-linear least-squares estimation.

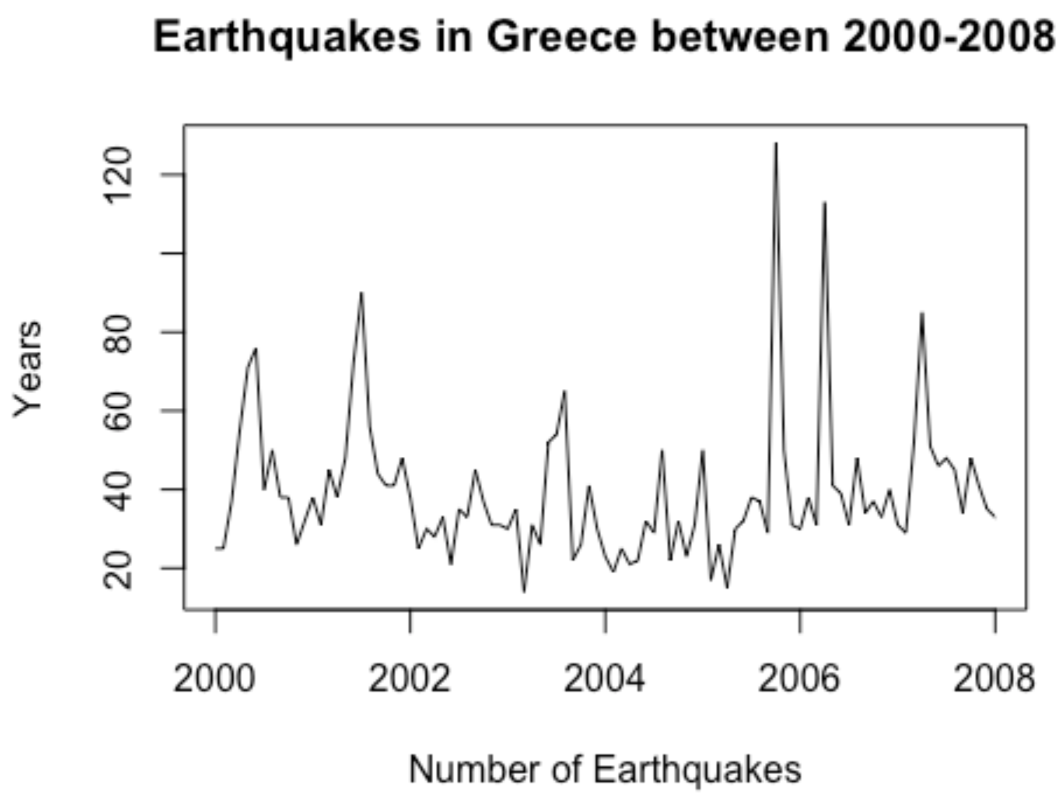
Model checking by testing whether the estimated model conforms to the specifications of a stationary univariate process. In particular, the residuals should be independent of each other and constant in mean and variance over time. (Plotting the mean and variance of residuals over time and performing a Ljung-Box test or plotting autocorrelation and partial autocorrelation of the residuals are helpful to identify misspecification.) If the estimation is inadequate, we have to return to step one and attempt to build a better model.

Time series analysis in R

The ts() function will convert a numeric vector into an R time series object. The format is ts(vector, start=, end=, frequency=) where start and end are the times of the first and last observation and frequency is the number of observations per unit time (1=annual, 4=quartly, 12=monthly, etc.).

```
seismic <- count(data, c("date", "mo"))

# set the freq parameter to 12 to indicate monthly readings
# ts() function to create a new time series
seismic_timeseries <- ts(seismic$freq, start = c(2000, 1), end =
c(2008, 1), frequency = 12)
plot(seismic_timeseries, xlab='Number of Earthquakes' ,ylab='Years',
main='Earthquakes in Greece between 2000-2008')
```




```
## -------------   USE ARIMA MODEL  ---------------------
#creating ranges of possible values for the order parameters p, d, and
q.
d <- 0 : 2
p <- 0 : 6
q <- 0 : 6
```

```r
seismic_models <- expand.grid(d = d, p = p, q = q)
head(seismic_models, n = 4)
```

```
##   d p q
## 1 0 0 0
## 2 1 0 0
## 3 2 0 0
## 4 0 1 0
```

```r
getTSModelAIC <- function(ts_data, p, d, q) {
                                            ts_model <- arima(ts_data,
order = c(p, d, q))
                                              return(ts_model$aic)
                                              }


getTSModelAICSafe <- function(ts_data, p, d, q) {
                                                 result = tryCatch({
                                                 getTSModelAIC(ts_data,
p, d, q)
                                                  }, error =
function(e) {
                                                 Inf
                                                 })
                                                 }

 # PICK THE BEST MODEL THAT HAS THE SMALLEST AIC
seismic_models$aic <- mapply(function(x, y, z)
                              getTSModelAICSafe(seismic_timeseries, x,
y, z), seismic_models$p,
                              seismic_models$d, seismic_models$q)
```

```
## Warning in log(s2): NaNs produced

## Warning in arima(ts_data, order = c(p, d, q)): possible convergence
## problem: optim gave code = 1

## Warning in arima(ts_data, order = c(p, d, q)): possible convergence
## problem: optim gave code = 1

## Warning in log(s2): NaNs produced

## Warning in arima(ts_data, order = c(p, d, q)): possible convergence
## problem: optim gave code = 1

## Warning in arima(ts_data, order = c(p, d, q)): possible convergence
## problem: optim gave code = 1

## Warning in arima(ts_data, order = c(p, d, q)): possible convergence
## problem: optim gave code = 1
```

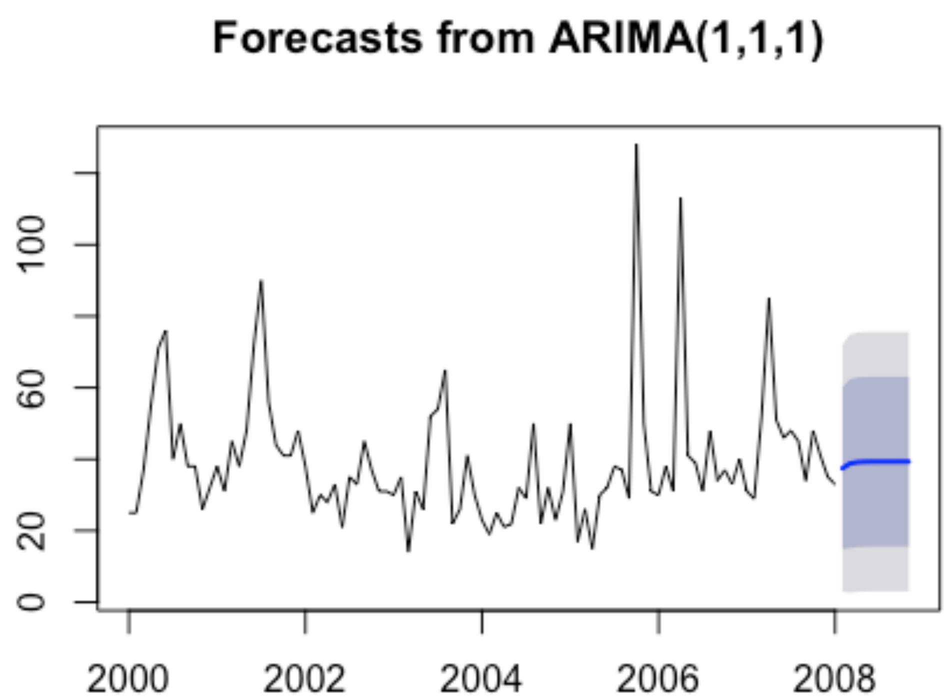
```r
subset(seismic_models,aic == min(aic))

##    d p q     aic
## 26 1 1 1 832.171

# ARIMA model for best p,d,q order model
seismic_model <- arima(seismic_timeseries, order = c(1, 1, 1))
summary(seismic_model)

##
## Call:
## arima(x = seismic_timeseries, order = c(1, 1, 1))
##
## Coefficients:
##          ar1      ma1
##       0.2949  -1.0000
## s.e.  0.0986   0.0536
##
## sigma^2 estimated as 306.9:  log likelihood = -413.09,  aic = 832.17
##
## Training set error measures:
##                      ME     RMSE      MAE       MPE     MAPE
MASE
## Training set -0.2385232 17.42922 11.12018 -14.47481 29.84171
0.8174096
##                     ACF1
## Training set -0.02179457

#-------------------       Prediction    ------------------------------
---
plot(forecast(seismic_model, 10))
```



Resources

- Constants and ARIMA models in R | Hyndsight
- Time Series ARIMA Models in R
- Identifying Seasonal Models and R Code | STAT 510
- R Time Series Tutorial
  # References

The data, R code and lessons are based upon:

1. Time Series Analysis :

Data Source: http://www.geophysics.geol.uoa.gr/catalog/catgr_20002008.epi

Code References :

Book : Mastering Predictive Analytic with R
Author: Rui Miguel Forte
https://www.safaribooksonline.com/library/view/mastering-predictive-analytics/9781783982806/

Chapter 9: Time series Analysis

http://www.statoek.wiso.uni-goettingen.de/veranstaltungen/zeitreihen/sommer03/ts_r_intro.pdf

http://www.stat.pitt.edu/stoffer/tsa3/R_toot.htm

http://www.statoek.wiso.uni-goettingen.de/veranstaltungen/zeitreihen/sommer03/ts_r_intro.pdf

2. Trend Analysis

Code References :

Book : Mastering Predictive Analytic with R
Author: Rui Miguel Forte
https://www.safaribooksonline.com/library/view/mastering-predictive-analytics/9781783982806/

http://www.r-bloggers.com/seasonal-trend-decomposition-in-r/

3. Seasonal Models

Code references :

Book: Time Series Analysis and Its Applications
Author: Robert H. Shumway . David S. Stoffer
Link:
http://www.springer.com/us/book/9781441978646#otherversion=9781461427599

http://a-little-book-of-r-for-time-series.readthedocs.org/en/latest/src/timeseries.html

https://onlinecourses.science.psu.edu/stat510/?q=node/47

https://rpubs.com/ryankelly/tsa5

https://onlinecourses.science.psu.edu/stat510/node/68

Data Reference :
https://github.com/RMDK/TimeSeriesAnalysis/blob/master/colorado_river.csv

4. Spectral Analysis

Code References:
Book:
Modern Applied Statistics with S Fourth edition
Author: W. N. Venables and B. D. Ripley
Link: Modern Applied Statistics with S Fourth edition

http://www.maths.adelaide.edu.au/patty.solomon/TS2004/tsprac3_2004.pdf