

Extract data from IRIS

Jonathan M. Lees
University of North Carolina, Chapel Hill
Department of Geological Sciences
CB #3315, Mitchell Hall
Chapel Hill, NC 27599-3315
email: jonathan.lees@unc.edu
ph: (919) 962-1562

January 23, 2020

Short tutorial for extracting data from IRIS. There are 2 basic routine packages, here is some R code:

```
> options(continue=" ")
> options(width=60)
> options(prompt=" ")
options(continue=" ")
options(SweaveHooks=list(fig=function()
par(mar=c(5.1, 4.1, 1.1, 2.1))))
library(RPMG)
nomoredev<-function()
{
  dev.off()
}
```

Set up the libraries:

```
library(IRISseismic)
library(rFDSN) #This is on CRAN
library(lubridate)
library(maps)
library(maptools)
```

Here is how it works:

```
## iris <- new("IrisClient")
## starttime <- as.POSIXct("2014-05-01",tz="GMT")
## endtime <- starttime + 30*24*3600

## TAstations <- getStation(iris,"TA","*" ,"*","BHZ",starttime,endtime )

glim=list()
glim$x=c(-85.5280721906,-65.8755069091)
glim$y=c(33.3917838213,47.8802768649)
iris <- new("IrisClient")
starttime <- as.POSIXct("2015-03-01",tz="GMT")
endtime <- as.POSIXct("2015-03-31",tz="GMT") + 24*3600
TAstations <- IRISseismic::getStation(iris,"*","*","*", "BDF",
```

```

                                starttime,
                                endtime)
jpeg(file="./FIGS/allstations_array.png", width=8, height=8)
map('usa')
map('state', add=TRUE)
points(TAstations$longitude, TAstations$latitude )
## GL = locator()
GL=list()
GL$x=c(-83.4794370355,-80.6579065573,
       -75.0148456010,-72.1933151229,-74.6829008389,
       -80.2429756047,-83.9773541787)
GL$y=c(37.6643393197,37.1624821297,
       37.4761428734,40.1108931209,43.6238934508,
       44.0002863433,41.6164646909)
polygon(GL, border='red')
dev.off()
  JJpoly = splancs::as.points(GL$x, GL$y)
JJpoints = splancs::as.points( TAstations$longitude,TAstations$latitude)
  IN = splancs::inout(JJpoints,JJpoly,bound=NULL,quiet=TRUE)
jpeg(file="./FIGS/restricted_array.png", width=8, height=8)
map('usa')
map('state', add=TRUE)
points(TAstations$longitude[IN],TAstations$latitude[IN] )
dev.off()

```

1 Streaming Traces

```

source("~/Site/TA_DATA/CODE/stream2GH1.R")
## print(stream2GH1)

```

First find and event on the IRIS catalog site.

```

library(IRISseismic)
###library(lubridate)

iris <- new("IrisClient")
starttime <- as.POSIXct("2015-02-12 05:00:00",tz="GMT")
endtime <- as.POSIXct("2015-02-12 06:00:00",tz="GMT")
#### find data that has infrasound
TAstations <- IRISseismic::getStation(iris,"TA","*","*", "BDF",
                                     starttime,
                                     endtime)

#### choose one

avail = vector(mode='logical', length=length(TAstations$station))
for(ksta in 1:length(TAstations$station) )
{
  onesta = TAstations$station[ksta]
  gtestBDF= getAvailability(iris, 'TA', onesta, '*', 'BDF',
                           starttime, endtime)

  utest = unlist(gtestBDF)
  if(length(utest)<1) { avail[ksta] = FALSE } else {avail[ksta]=TRUE }
}
w1 = which(avail)

```

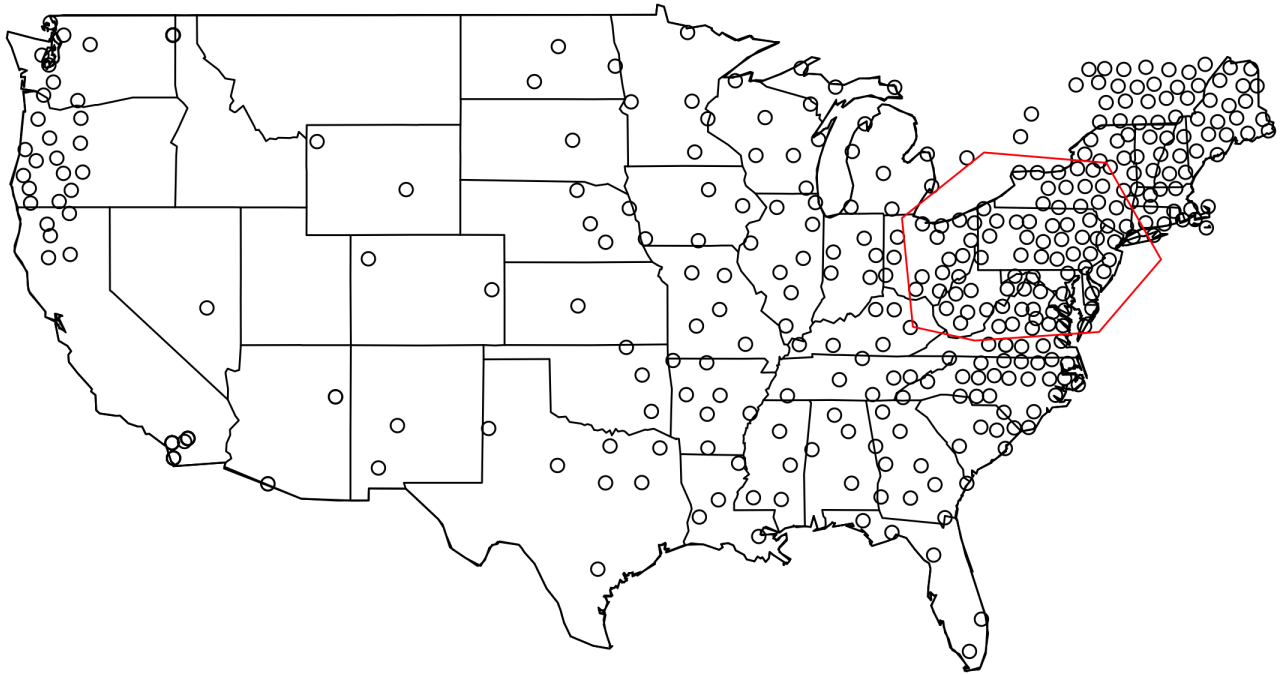


Figure 1: Restricted polygon for extracting data

```
ksta = w1[20]
onesta = TAstations$station[ksta]
P1 = IRISseismic::getDataselect(iris, "TA", onesta, '*', 'BDF',
                                starttime, endtime, '*')
P2 = IRISseismic::getDataselect(iris, "TA", onesta, '*', 'BHZ',
                                starttime, endtime, '*')
GG1 = stream2GH1(P1)
GG2 = stream2GH1(P2)
```

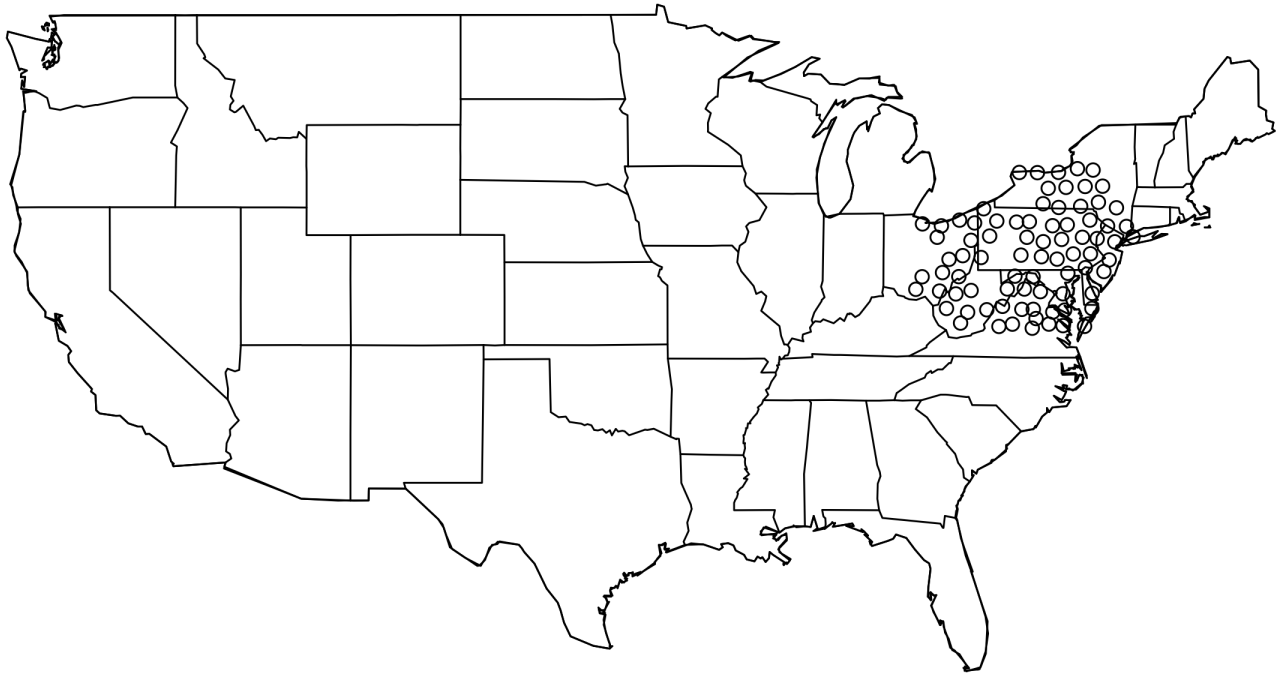


Figure 2: Restricted polygon for extracting data

1.1 Event extraction from IRIS

```
library(IRISseismic)
iris2 <- new("IrisClient")
starttime <- as.POSIXct("2015-03-01 00:00:00",tz="GMT")
endtime <- as.POSIXct("2015-03-12 06:00:00",tz="GMT")
events <- getEvent(iris2, starttime, endtime, minmag=5.0)
LENE = length(events$eventID)
we = which.max(events$magnitude)
starttime <- as.POSIXct(events$time[we] ,tz="GMT")-10
```

BGNE BHZ 2015-03-10 20:55:34

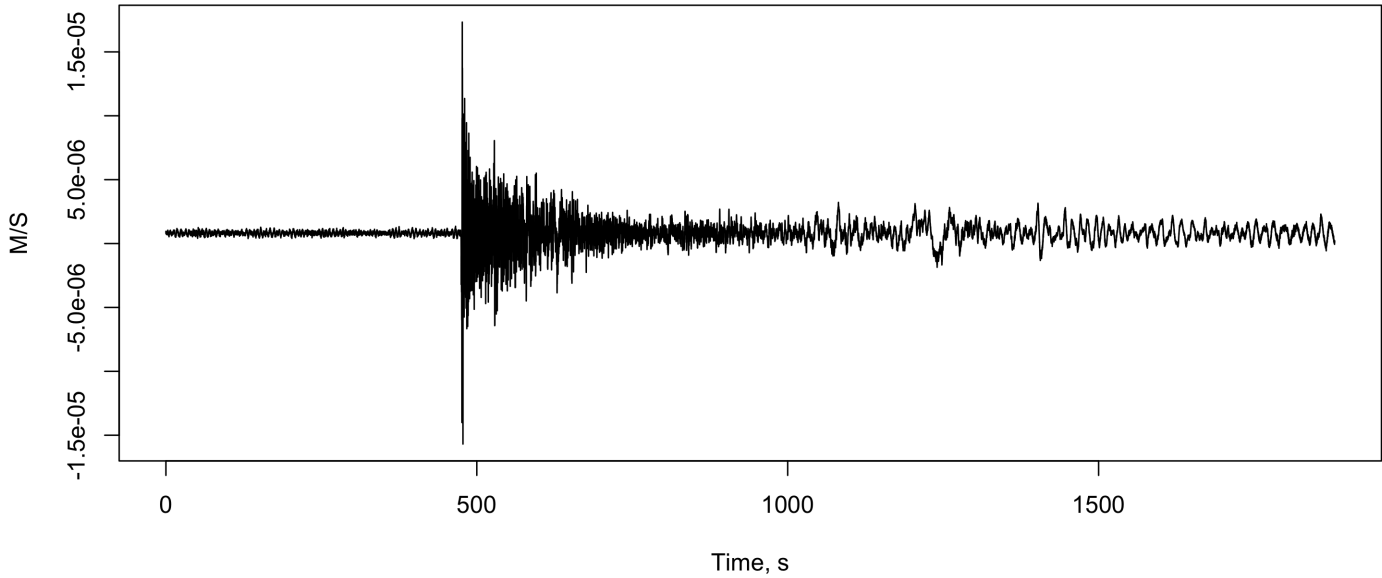


Figure 3: One wiggle using RSEIS and IRISseismic packages

```
endtime <- starttime + 1800
Tastations <- IRISseismic::getStation(iris2,"TA","*","*","BHZ",
                                     starttime,
                                     endtime)

asta = Tastations$station[11]
chan = 'BHZ'
P2 = getDataselect(iris2, 'TA', asta, '*', chan,
                  starttime, endtime, '*')
#### extrac tthe streaming data and put into RSEIS style format
H1 = stream2GH1(P2, DEST='.', STREAM=TRUE )
jpgn(file='./FIGS/OneWig.png', width=10, height=5)
plot(seq(from=0, by=H1[[1]]$dt, length=H1[[1]]$N) , H1[[1]]$amp, type='l', ylab='M/S', xlab='Time, s')
title(main=paste(asta, chan, starttime) )
dev.off()
```