SingleFacility.R

Connor Brogan

March 11, 2018

## Summary

The goal of this code is to extract discharge data from Discharge Monitoring Records (DMR) for a single facility within a given state using the EPA’s Enforcement and Compliance History Online (ECHO) RESTful services. The code will first download a list of active and inactive facilities across the state. Then, a user may select a single facility using ECHO’s *SourceID* to download associated DMR data. SingleFacility.R automatically limits the DMR to discharge data, but users may change the *parameter\_code* to fit their desired metrics.This code has been written into a function and is avialble at the bottom of the script to be called repeatedly for easy analysis of DMRs. Scripts ECHOInterface.R and AnalysisCode.R have been written to perform similar analysis on a state-wide basis. This script has only been tested for Virginia. Although it likely loads data for other states, data completeness and quality should first be reviewed on the ECHO webpage [here](https://echo.epa.gov/resources/echo-data/about-the-data#completeness).

## Data Initialization

SingleFacility.R begins by calling relevant libraries. It uses *XML* and *RCurl* to facilitate downloads, parsing, and formatting of ECHO data. These libraries are largely used to run ECHO queries and hande returned data.

library(XML)  
library(RCurl)

In this section, users must define their **state** of interest using it’s two letter code (demonstration is for Virginia, VA) and a **date range** of interest for DMR statistics. ECHO is limited to the past three years and any date outside of that range will be defaulted by ECHO to available data within the three year period. Optional lines are commented out using the pound sign that will automatically fill the end date with the current computer system date (these can be activated by deleting the pound sign).

state<-"VA"  
startDate<-"01/01/2016"  
endDate<-"12/31/2016"  
#endDate<-Sys.Date()  
#endDate<-format(as.Date(endDate), "%m/%d/%Y")

## Execution

This portion of the script queries ECHO to develop a data frame *a* containing all active and inactive effluent monitoring facilities across the state. This list is not exclusive to those monitoring discharge under the Clean Water Act, but instead represent a list of all facilities that are or have previously been moniroted under federal regulations include the CWA, CAA, SWDA, and RCRA. The facilities have a name, ECHO identifier (*SourceID*), and a physical adress.

uri\_query<-paste0("https://ofmpub.epa.gov/echo/cwa\_rest\_services.get\_facilities?output=XML&p\_st=",state,"&p\_tribedist=0")  
ECHO\_xml<-getURL(uri\_query)  
ECHO\_query<-xmlParse(ECHO\_xml)  
QID<-xmlToList(ECHO\_query)  
QID<-QID$QueryID  
uri\_summary<-paste0("https://ofmpub.epa.gov/echo/cwa\_rest\_services.get\_download?output=CSV&qid=",QID)  
a<-read.csv(uri\_summary,stringsAsFactors = F)

## Observations: 1,952  
## Variables: 10  
## $ CWPName <chr> "ABB INCORPORATED - BLAND", "ABINGDON R...  
## $ SourceID <chr> "VA0086355", "VAL026531", "VA0027162", ...  
## $ CWPStreet <chr> "171 INDUSTRY DR", "21436 VANCE MILL RO...  
## $ CWPCity <chr> "BLAND", "ABINGDON", "MAPPSVILLE", "MEL...  
## $ CWPState <chr> "VA", "VA", "VA", "VA", "VA", "VA", "VA...  
## $ CWPStateDistrict <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,...  
## $ CWPEPARegion <int> 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, ...  
## $ FacDerivedHuc <int> 5050002, 6010102, 2080109, 2080109, 208...  
## $ FacIndianSpatialFlg <chr> "N", "N", "N", "N", "N", "N", "N", "N",...  
## $ CWPTotalDesignFlowNmbr <dbl> 0.010, NA, NA, NA, NA, NA, NA, NA, NA, ...

No latitude and longitude coordinates are available for the facilities in this code. An optional secondary query can be run to give facility coordinates, as shown below. This will loop through each facility within **a** and add facility coordiantes. It requires the *jsonlite* package, which facilitates the download and formatting of javascript object notation. Note that coordinates are associated with a NAD 83 datum. Please note that this download will take a while to complete and it is recommended that you first narrow the list of facilities to those necessary in the intended geospatial analysis.

library(jsonlite)  
for (i in 1:length(a$CWPName)){  
 json\_file<-paste0("https://ofmpub.epa.gov/echo/dfr\_rest\_services.get\_dfr?output=JSON&p\_id=",a$SourceID[i])  
 json\_data<-fromJSON(txt=json\_file)  
 if(length(json\_data$Results$SpatialMetadata$Latitude83)>0){  
 a$Faclat[i]<-json\_data$Results$SpatialMetadata$Latitude83  
 a$Faclong[i]<-json\_data$Results$SpatialMetadata$Longitude83  
 } else {  
 a$Faclat[i]<-NA  
 a$Faclong[i]<-NA  
 }  
}

Upon review of data frame *a*, users may input a facility *SourceID* below to download DMR data associated with that facility for the given date range. As an example, the below code shows some of the DMR data for the Surry Power Station, ECHO source identifier VA0004090. Note how the identifier is input as charater data using quotations.

sourceID<-"VA0004090"  
uri\_effluent<-paste0("https://ofmpub.epa.gov/echo/eff\_rest\_services.download\_effluent\_chart?p\_id=",sourceID,"&start\_date=",startDate,"&end\_date=",endDate)  
facility<-read.csv(uri\_effluent,stringsAsFactors = F)  
facility<-facility[facility$parameter\_code==50050,]

## Observations: 150  
## Variables: 10  
## $ activity\_id <dbl> 3200118042, 3200118042, 3200118042, 320...  
## $ npdes\_id <chr> "VA0004090", "VA0004090", "VA0004090", ...  
## $ version\_nmbr <int> 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...  
## $ perm\_feature\_id <dbl> 3200037830, 3200037830, 3200037830, 320...  
## $ perm\_feature\_nmbr <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...  
## $ perm\_feature\_type\_code <chr> "EXO", "EXO", "EXO", "EXO", "EXO", "EXO...  
## $ perm\_feature\_type\_desc <chr> "External Outfall", "External Outfall",...  
## $ limit\_set\_id <dbl> 3200046298, 3200046298, 3200046298, 320...  
## $ limit\_set\_schedule\_id <dbl> 3200092597, 3200092597, 3200092597, 320...  
## $ limit\_id <dbl> 3401468827, 3401468827, 3401468827, 340...

## Function for Easy Use

At the end of SingleFacility.R is a function that can be called elsewhere for easy use to draw in DMR data. It outputs the DMR data frame and takes in a start date, end date, source ID, and parameter code. It assumes discharge data is desired and has a default parameter code of 50050 as such. An example of its use is shown below:

DMR<-function(sourceID,startDate,endDate,parameter\_code=50050){  
uri\_effluent<-paste0("https://ofmpub.epa.gov/echo/eff\_rest\_services.download\_effluent\_chart?p\_id=",sourceID,"&start\_date=",startDate,"&end\_date=",endDate)  
facility<-read.csv(uri\_effluent,stringsAsFactors = F)  
facility<-facility[facility$parameter\_code==parameter\_code,]  
return(facility)  
}  
exampleDMR<-DMR(sourceID = "VA0004090",startDate = "01/01/2016",endDate = "12/31/2016",parameter\_code = 50050)

## Observations: 150  
## Variables: 10  
## $ activity\_id <dbl> 3200118042, 3200118042, 3200118042, 320...  
## $ npdes\_id <chr> "VA0004090", "VA0004090", "VA0004090", ...  
## $ version\_nmbr <int> 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...  
## $ perm\_feature\_id <dbl> 3200037830, 3200037830, 3200037830, 320...  
## $ perm\_feature\_nmbr <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...  
## $ perm\_feature\_type\_code <chr> "EXO", "EXO", "EXO", "EXO", "EXO", "EXO...  
## $ perm\_feature\_type\_desc <chr> "External Outfall", "External Outfall",...  
## $ limit\_set\_id <dbl> 3200046298, 3200046298, 3200046298, 320...  
## $ limit\_set\_schedule\_id <dbl> 3200092597, 3200092597, 3200092597, 320...  
## $ limit\_id <dbl> 3401468827, 3401468827, 3401468827, 340...