The SampleSplitting R package

Jessica Thompson[1], Laura Hubbard[1], Matt Komiskey[1], and Todd Stuntebeck[1]

[1] *United States Geological Survey*

November 10, 2015

Contents

1 Introduction to SampleSplitting

SampleSplitting: An R package for the automation of splitting time-paced runoff event samples, including creation of summary table for laboratory use.

The best way to learn about the time-paced runoff event sampling technique is to read the USGS Open-file Report that summarizes data collection, processing and analysis for the Discovery Farms and Pioneer Farm sites in Wisconsin. The url for ((1) Stuntebeck, et al. 2008) is `http://pubs.usgs.gov/of/2008/1015/pdf/ofr2008-1015.pdf`

This vignette assumes that the user understands the basics of runoff event sampling and the USGS ADAPs system. The vignette will walk through an example workflow using provided data for a real storm event. An example workflow script is also provided online at `https://github.com/USGS-R/SampleSplitting/tree/master/inst` and in your local R package library.

This package also requires the use of the packages googleVis and dataRetrieval. The googleVis package provides an interface between the Google Charts API and R, for the creation of interactive charts. The dataRetrieval package enables easy retrieval of USGS data from available web services or user provided data files.

For information on getting started in R and installing the package, see (3):Getting Started in R.

2 General Workflow

This example vignette first loads the relevant packages (assuming they are already installed) and the included sample data. In your use, you would need to load the packages, but not the given example data. First, it is a good practice to clear any existing objects from your working environment. In RStudio, in the upper right quadrant, there is a broom icon that says "Clear". Click this and select "Yes" to remove objects. This allows you to being work with a clear workspace, negating the possibility of mis-identified variables.

```
library(SampleSplitting)
library(dataRetrieval)
library(googleVis)

##############################################
# Load sample data included with package:
ExampleData <- rdbExample
```

Enter information about the USGS station, date range and optional separate USGS station for precipitation data.

```
##############################################
# Enter information about the desired data set
# enter NWIS station id for gaging station
siteNo <- "424314090240601"
# enter date to begin pulling data (rounded to the day)
StartDt <- '2008-05-30'
# enter date to stop pulling data (rounded to the day)
EndDt <- '2008-06-15'
# enter NWIS station id for precipitation gaging station, may or
# may not be identical to "siteNo"
precipSite <- "434425090462401"
```

Normally, the user will now run the getADAPSData function with or without a dataFile variable, depending on if a data file exported from ADAPs is being used. For this example, we will simply rename the provided data.

```
################################################
# rename example data for following operations
adaps_data_all <- ExampleData
```

The retrieved data is now saved as a comma-separated text file, in case you need to refer to it later. The file will be saved to your present working directory. If you are unsure of that location, you can run: "getwd()" at the R prompt to discover this information.

```
##############################################
# save merged data for station/storm event, saved as file,
# eg 434425090462401data.csv
```

```
mergedDataTable(siteNo,StartDt,EndDt,adaps_data_all)
```

Now the hydrologist may generate hydrographs to assist in choosing storm event start and end date-times. There are two hydrographs available, one which generates an interactive Google charts visualization, which will pop up in a browser, and one that saves as a pdf to your working directory.

```
# Generate interactive googleVis plot
hydrographPlot <- hydrographInteractive(adaps_data_all)
plot(hydrographPlot)

# Generate pdf of hydrograph to save, saved as file,
# eg 434425090462401hydrograph.pdf
# adjust dateInt as desired to vary length of time (in hours) between
# x-axis tick marks
hydrographPDF(adaps_data_all,siteNo,dateInt=8)
```

Now that storm start and end times have been determined, it is time to enter information for each event. You must enter a StormStart, StormEnd and StormName for each event. For sampled events, you must also enter maxBottleVol, maxSampleVol and subNum. These values must be in order (eg, all lists must be in the same order as the StormName list). If you have unsampled storms, just skip the maxBottleVol, maxSampVol and subNum for those. So, if I have 3 storm events - "S1-01","S1-02","S1-03" - and S1-02 wasn't sampled, my lists might be:

```
StormName      c("S1-01","S1-02","S1-03")
StormStart     c("2012-04-01 14:00","2012-04-02 20:30","2012-04-05 10:15")
StormStart     c("2012-04-01 22:30","2012-04-03 02:15","2012-04-05 13:45")
maxBottleVol   c(400,400)
maxSampVol     c(3900,3900)
subNum         c(1,5)
```

```
############################################################
# after using the hydrographs to determine storm start and end time(s),
# enter information for storm events.
# IF you have un-sampled storms, you may enter their StormStart and
# StormEnd values, as well as StormNames in
# the appropriate list. Leave them out of the maxBottleVol, maxSampVol
# and subNum lists
# enter the name of the storm event(s)
StormName <- c("S2-066","S2-066A","S2-067","S2-068","S2-069",
               "S2-070","S2-071","S2-072","S2-073","S2-074",
               "S2-075")
# enter Storm Start date(s)
# MUST be in the format YYYY-MM-DD HH:24
StormStart <- c("2008-05-30 02:51","2008-06-01 02:30",
                "2008-06-05 04:39","2008-06-06 04:22",
                "2008-06-07 22:52","2008-06-08 08:41",
```

```
                "2008-06-08 19:03","2008-06-12 09:03",
                "2008-06-12 21:40","2008-06-14 16:52",
                "2008-06-15 04:07")
# enter Storm End date(s)
# MUST be in the format YYYY-MM-DD HH:24
StormEnd <- c("2008-05-30 08:49","2008-06-01 22:45",
              "2008-06-05 07:21","2008-06-06 05:28",
              "2008-06-08 01:14","2008-06-08 11:39",
              "2008-06-08 21:31","2008-06-12 10:22",
              "2008-06-13 01:36","2008-06-14 18:05",
              "2008-06-15 09:22")
# enter the maximum possible volume for one sample bottle
maxBottleVol <- c(400,600,600,600,600,600,600,400,600,800)
# enter the maximum possible volume for one full storm sample
maxSampVol <- c(3900,3900,3900,3900,3900,3900,3900,3900,3900,3900)
# enter number for 1st bottle of each storm, if a number other
# than 1 is desired
subNum <- c(1,1,1,1,16,1,1,5,1,7)
```

Once that information has been entered, it is time to run the calculation for sample splitting volumes. After the calculation has been done, it may be helpful to output the initial numbers for review, as well as save the intermediate volume calculations (for later reference).

```
# generate bottle volume table(s) for lab for each storm
tableOut <- labDataOut(adaps_data_all,StormStart,StormEnd,StormName,
                       maxBottleVol,maxSampVol,subNum=subNum)
# look at table(s) generated for lab sample instructions for storm
# event(s) and determine if changes are needed
for (i in 1:length(StormStart)){
  print(tableOut[[i]])
}

#Output csv file of all intermediate volumes used for calculations
intermediateVolTable(siteNo,StormStart,StormEnd,tableOut)
```

After looking at the initial volumes, if there is a sample that needs to be removed, you can optionally run the following section to remove the desired sample and also re-calculate volumes and save the updated information.

```
# OPTIONAL if sample(s) need to be removed, enter their datetime
# and a comment and re-create tableOut
# MUST be in the format YYYY-MM-DD HH:24
removeDate <- c("2008-05-30 07:44")
removeComment <- c("")
tableOut <- labDataOut(adaps_data_all,StormStart,StormEnd,StormName,
                       maxBottleVol,maxSampVol,
```

```
                          removeDate=removeDate,subNum=subNum)
for (i in 1:length(StormStart)){
  print(tableOut[[i]])
}
intermediateVolTable(siteNo,StormStart,StormEnd,tableOut)
```

When you are satisfied with the sample splitting volume calculations, it is time to enter the date when sample bottles were retrieved and generate two summary files. The first summary file is named, for example, S2-066sampVol.txt. It contains the volume calculations for the sample bottles for all storm events as well as extra information for the hydrologist, such as the total volume fo the lab sample and the total sampled storm volume. The other file, S2-066labVolumes.txt, contains a clean table of bottle volumes for each storm event to be sent to the lab.

```
#Once you are satisfied with the table output
#enter date(s) when samples were picked up
bottlePickup <- c("Bottles S2-1 through S2-26 picked up 2008-05-30",
                  "Bottles S2-1 through S2-15 picked up 2008-06-05",
                  "Bottles S2-1 through S2-6 picked up 2008-06-06",
                  "Bottles S2-1 through S2-9 picked up 2008-06-07 and
                  bottles S2-10 through S2-15 picked up 2008-06-08",
                  "Bottles S2-16 through S2-25 picked up 2008-06-09",
                  "Bottles S2-1 through S2-9 picked up 2008-06-10",
                  "Bottles S2-1 through S2-7 picked up 2008-06-12",
                  "Bottles S2-5 through S2-25 picked up 2008-06-13",
                  "Bottles S2-1 through S2-6 picked up 2008-06-14",
                  "Bottles S2-7 through S2-17 picked up 2008-06-15")

# generate text file with storm event sample bottle
# volume table(s)
stormEventsTable(StormName,StormStart,StormEnd,tableOut,
                 maxBottleVol,bottlePickup)

# generate simple table for lab
labVolumesTable(StormName,StormStart,StormEnd,tableOut,
                bottlePickup)
```

3 Getting Started in R

This section describes the options for downloading and installing the sampleSplitting package.

3.1 New to R?

If you are new to R, you will need to first install the latest version of R, which can be found here: `http://www.r-project.org/`. There is also a useful USGS site for R help at `http://bwtst.usgs.gov/apps/R/index.html`.

There are many options for running and editing R code, one nice environment to learn R is RStudio. RStudio can be downloaded here: `http://rstudio.org/`. Once R and RStudio are installed, the dataRetrieval package needs to be installed as described in the next section.

At any time, you can get information about any function in R by typing a question mark before the functions name. This will open a file (in RStudio, in the Help window) that describes the function, the required arguments, and provides working examples.

```
library(SampleSplitting)
?getADAPSData
```

To see the raw code for a particular code, type the name of the function:

```
getADAPSData
```

Additionally, many R packages have vignette files attached (such as this paper). To view the vignette:

```
vignette(SampleSplitting)
```

3.2 R User: Installing SampleSplitting

Before installing SampleSplitting, the supporting packages must be first be installed:

```
install.packages(c("googleVis"), dependencies=TRUE)
install.packages(c("dataRetrieval"), repos="http://usgs-r.github.com",type="sou
install.packages(c("SampleSplitting"),repos="http://usgs-r.github.com",
                 type="source")
```

It is a good idea to re-start R after installing the package, especially if installing an updated version. Some users have found it necessary to delete the previous version's package folder before installing newer version of dataRetrieval. If you are experiencing issues after updating a package, trying deleting the package folder - the default location for Windows is something like this: C:/Users/userA/Documents/R/win-library/2.15/dataRetrieval, and the default for a Mac: /Users/userA/Library/R/2.15/library/dataRetrieval. Then, re-install the package using the directions above. Moving to CRAN should solve this problem.

After installing the package, you need to open the library each time you re-start R. This is done with the simple command:

```
library(SampleSplitting)
```

References

[1] Stuntebeck, T.D., Komiskey, M.J., Owens, D.W., and Hall, D.W., 2008, Methods of data collection, sample processing, and data analysis for edge-of-field, streamgaging, subsurface-tile, and meteorological stations at Discovery Farms and Pioneer Farm in Wisconsin, 2001-7: U.S. Geological Survey Open-File Report 2008-1015, 51 p. http://pubs.usgs.gov/of/2008/1015/pdf/ofr2008-1015.pdf