RWater Module 3

Understanding Rainfall-Streamflow Relationship from Real-time Gage Station Data

Adnan Rajib and Venkatesh Merwade

Lyles School of Civil Engineering, Purdue University

Learning Goals

From the hypothetical examples shown in the previous module, students already have the concepts of rainfall hyetographs and streamflow hydrographs, along with relevant terminologies such as Peak Discharge and Lag Time. The United States Geological Survey (USGS) has thousands of gage stations all over the USA, monitoring the streamflow. After completing this module, students will be able to:

i. download the USGS daily streamflow data for particular locations directly through RWater
ii. plot rainfall hyetograph and streamflow hydrograph by writing simple programming lines in RWater interface, based on the actual data at any USGS location
iii. visualize the effect of rainfall intensity over lag time and peak discharge in an interactive way using the graphs created by their own

The script for this module is given below. Since this module is interactive, the given script has to be written/edited in RWater interface and run in steps *, **. Relevant explanations associated with each step are also provided here (lines followed by # sign). These explanatory lines are only for building user's perception over the code and hence, not required to write in RWater.

Downloading the USGS Streamflow Data

The objective for this portion of the script is to enable direct downloading of USGS daily streamflow data from any valid location and for any valid date range. The downloaded data can be used for subsequent plotting and analysis without any sort of pre-processing. In this module, we are going to use the data from USGS 04180000 which is a gage station for the watershed called Cedar Creek near Cedarville in Indiana, USA. Click on http://goo.gl/Y5lYZZ and you can see the watershed as well as the gage station in a customized Google map.

STEP 1

```
cat("\014")
rm(list = ls())
dev.off()

library(waterData)
```

Removing all previously generated datasets and plots

Loading the **waterData** package (case sensitive: **D** is capital letter). A package is a collection of R functions to serve specific purpose

STEP 2

```
site<-c("XXXX")
cedarinfo<-siteInfo(site)
```

Getting information about a USGS site. In this example, replace **XXXX** with **04180000**. This is the USGS gage station ID for Cedar Creek watershed in Indiana

siteInfo is the function name which pulls the information from USGS website (case sensitive: **I** is capital letter)

Click on 'cedarinfo' in the workspace (upper right).
You can now see the name, latitude and longitude of the station

After loading the Module 3 script from RWater working directory, make changes only in the shaded portion

****Run in steps***

```
cedarflow<-importDVs("XXXX",
                     code="00060", stat="00003",
                     sdate="YYYY-MM-DD",
                     edate="YYYY-MM-DD")
```

Downloading streamflow data directly from USGS. In this example, replace **XXXX** with **04180000**.

importDVs is the function name which pulls the data from USGS website (case sensitive: **D** and **V** are capital letters)

After running this step, the downloaded data gets saved with a name **cedarflow**. Click on 'cedarflow' in the workspace (upper right). You can see that streamflow values are listed in the right column (column header name is **val**)

"00060" and "00003" indicates 'Streamflow' and 'Mean Daily Data' respectively

Provide a range of date within **sdate** and **edate** in YYYY-MM-DD format. In this example, use 2009-01-01 and 2010-12-31 respectively

Loading a Given Rainfall Dataset from RWater Working Directory

The most comprehensive source of daily precipitation data is on the website of the National Climatic Data Center (NCDC). To aid the easy usage of this module, we have prepared the rainfall dataset for Cedar Creek watershed, which you can find in your RWater working directory.

STEP 3

```
cedarrain=read.csv("/home/mygeohub/
                   username/working directory name/
                   cedarrain.csv",header = TRUE,
                   stringsAsFactors = FALSE)
```

Load the given rainfall dataset from your working directory. File name is **cedarrain**

Use your own RWater **username** and the **name of your current working directory**

stringsAsFactors is case sensitive (**A** and **F** are capital letters). See Appendix for the meaning of the function

Click on 'cedarrain' in the workspace (upper right). You can now see the rainfall data set (two columns: one is date and the other one is the rainfall value in mm)

Plotting Rainfall Hyetograph and Streamflow Hydrograph

STEP 4

```
par(mfrow=c(1,1))
```

Option of creating a multi-paneled plotting window (see Appendix for details). Here, we will have only one figure in the Plots window

STEP 5

```
par(mar=c(5, 4, 4, 8) + 0.1)
barplot(cedarrain$Rainfall.mm.,cedarrain$Date,
        space = c(0,1),width = 0.5,
        ylim=rev(c(0,XXXX)),
        xlab="Days in year 2009-2010",
        ylab="Rainfall(mm)",
        main="Cedar Creek near Cedarville, IN",
        axes=TRUE,las=1,xaxt="n",
        col="light blue", border="light blue")
```

Plotting the rainfall values in the 'cedarrain' file as a bar diagram (**barplot** function)

Put the maximum limit of the Y axis in **ylim**. Use 200 for this example. **rev** indicates that values are to be plotted in a reverse Y axis

See Appendix for all other functions (example: xlab, ylab, main etc.)

```
par(new=T)
plot(cedarflow$val,type="l",pch=21,
     col="red", lty=12,lwd=1.5,
     yaxt="n", ylim=c(0,XXXX),
     xlab="",ylab="",axes=T)
axis(side=4)
mtext("Streamflow(cfs)",side=4, cex.lab=1,
      las=3,line=3, col="black")
legend(300,5500,"Streamflow", col = "red",
       lwd=1.5, lty=12, bty="n")
legend(300,4500,"Rainfall", col = "light blue",
       lwd=7, lty=1, bty="n")
border=c("black")
```

par(new=T) will plot the new graph over the previous graph

cedarflow$val commands RWater to plot only the streamflow values from the **val** column in the cedarflow file

type indicates what type of graph should be drawn. Here 'l' (small letter L) means a line plot

Put the maximum limit of the Y axis. For this example, use 6500 in **ylim**

See Appendix for all other functions (example: lty, lwd, side etc.)

You have just created a graph showing real-time rainfall and streamflow data for the Cedar Creek watershed in Indiana! To read the graph, pick any day from X axis and trace a vertical line up to where it intersects the plotted hyetograph/hydrograph. Reading horizontally to the left and right, you can determine the rainfall and discharge of the stream for that date.

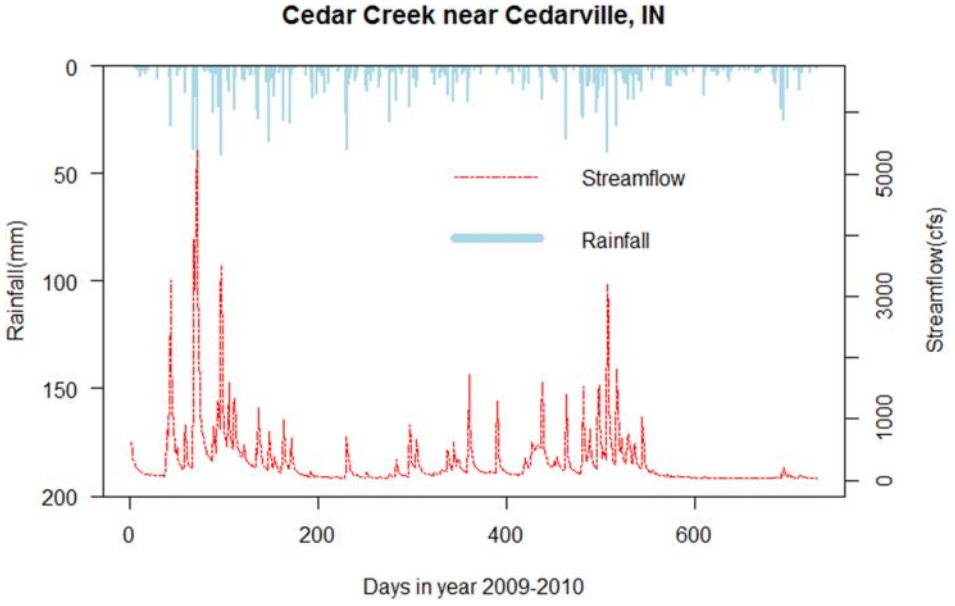

Figure 1

The graph which you have just plotted depicts vertical jumps and drops. What this means for the actual stream is that a stream exhibits increased flows immediately following the onset of a rainfall event and then return to pre-rain condition shortly after the end of rainfall.

Understanding Rainfall-Streamflow Relationship

In this step, we will split up the whole datasets being used in the previous step for a smaller duration of time. This will help to critically view the effect of rainfall intensity and temporal distribution over the streamflow response in an actual location such as Cedar Creek. This will also validate the theoretical concepts which we have discussed in Module 2, through a real-time situation.

STEP 6

```
library(xts)
cedarx<-subset(cedarflow,
               dates>="2009-05-25"
               &dates<="2009-06-04")
cedarrainx<-subset(cedarrain,
                   Date>="2009-05-25"
                   &Date<="2009-06-04")
```

Subsetting a date range directly from the original streamflow and rainfall datasets

We need an R package called **xts** for the subsetting task

The new subset datasets are named as '**cedarx**' and '**cedarrainx**'

STEP 7

```
par(mar=c(5, 4, 4, 8) + 0.1)
barplot(XXXX$Rainfall.mm.,XXXX$Date,
        space = c(0,1),width = 0.5,
        ylim=(c(0,XXXX)),
        xlab="Days (May 25 - June 4, 2009)", ylab="Rainfall(mm)",
        main="Cedar Creek near Cedarville, IN",
        axes=TRUE,las=1,xaxt="n",
        col="light blue", border="dark blue")
par(new=T)
plot(XXXX$val,type="l",pch=21,
     col="red", lty=12,lwd=1.5,
     yaxt="n", ylim=c(0,XXXX),
     xlab="",ylab="",axes=T)
axis(side=4)
mtext("Streamflow(cfs)",side=4, cex.lab=1,
      las=3,line=3, col="black")
legend(5,900,"Streamflow", col = "red",
       lwd=1.5, lty=12, bty="n")
legend(5,800,"Rainfall", col = "light blue",
       lwd=7, lty=1, bty="n")
border=c("black")
```

Use appropriate name of the data set from which **rainfall data** needs to be plotted (here, this should be the new subset **cedarrainx**)

Put the maximum limit of the Y axis. For this example, use 60 in **ylim**

Use appropriate name of the data set from which **streamflow data** needs to be plotted (here, this should be the new subset **cedarx**)

Put the maximum limit of the Y axis. For this example, use 1000 in **ylim**

All functions have the same meaning as in STEP 5