CSE Training – Data Analytics – Introduction to R – Project

This is a demo project for the CSE training course, Intro to R. We are going to use this project to demonstrate some basic data analysis that you can do in R. Specifically, you will learn the following about R: how to read data and clean data, different data types, useful functions and plotting systems.

In this project, we are going to do some data analysis using a data set downloaded from the National Climatic Data Center of National Oceanic and Atmosphere Administration. The data set is about the hourly precipitation in the Illinois state from July 1, 2012 to Jun 30, 2013.

Read Data

We read the data and take a quick look at the data using the following code.

```
# You want to change the working directory to where you place your data file.
# getwd() is a function to get the current working directory
# setwd() sets the the directory to the input, which is a character object
library(RCurl)
url <- "https://raw.githubusercontent.com/uiuc-cse/data-fa14/gh-pages/data/Illinois.csv";
file <- getURL(url, ssl.verifypeer=FALSE);

data = read.csv(textConnection(file), header = TRUE);# read.table()

str(data);# head(), tail()
```

Exercise

First download the data from the Internet. Then change the working directory to where you place the file. Use the function **read.csv()** to read data locally. If you feel unfamiliar with the function, use "?read.table" to get help.

Preprocess Data

In this project, we focus on precipitation on each day. So the hour information in the column *DATE* is not useful.

```
# as.character(), as.numeric, as.factor, etc.
data$DATE = as.character(data$DATE);

# gsub() substitute a certain pattern in a character with something else
temp = sapply(data$DATE, function(x) gsub(".{6}$", "",x));
data$DATE = as.Date(temp, "%Y%m%d");
```

Another thing we notice is that there are extreme values, like 99999, in the column *HPCP*. Then we look into the description file of the data set online and find that extreme value stands for missing values. So we substitute 99999 with NA.

```
data$HPCP[(data$HPCP==99999)] = NA;
```

Now we have our data set ready to use. We then move on to some basic analysis of the data.

Exercise

First, dig into the function **paste()**. Then run the following code.

```
a <- 123;
b <- "Hello, world!";
c <- as.character(a);
char <- paste(c, b, sep = "_");
```

Substitute all numbers in the variable *char* with asterisks.

Exploratory Data Analysis

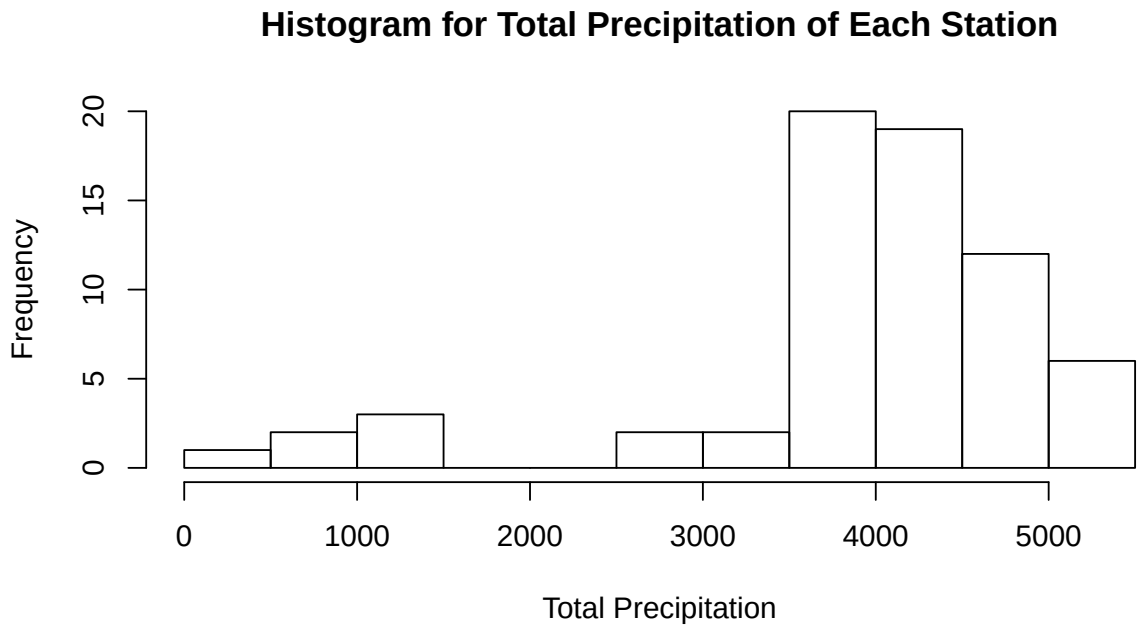
First, we would like to know, which station in Illinois has the largest precipitation during the observation period.

```
# split() splits an object according to another object
station_HPCP = sapply(split(data$HPCP, data$STATION_NAME), sum, na.rm = TRUE);

# data.frame()
station_HPCP = data.frame(station = names(station_HPCP), precipitation = station_HPCP);

# order()
station_HPCP = station_HPCP[order(station_HPCP$precipitation, decreasing = TRUE),];
rownames(station_HPCP) = c();

head(station_HPCP, n=5);
hist(station_HPCP$precipitation,
     main = "Histogram for Total Precipitation of Each Station",
     xlab = "Total Precipitation");
```



Second, we would like to see the pattern of total precipitation in Illinois through the observation period.

```
total_pre = sapply(split(data$HPCP, data$DATE), sum, na.rm = TRUE);
total_pre = data.frame(date = sort(unique(data$DATE)), value = total_pre);
rownames(total_pre) = c();

plot(total_pre$date, total_pre$value, type="l",
     xlab = "Date",
     ylab = "Precipitation",
     main = "Total Precipitation in Illinois");
```