Script Documentation

Language: Python

File Name: ReadR1R2.py

September 8, 2015

By:

Armen Arslanian

**Table of Contents**

* Introduction
* Prerequisites for the Script to Work Properly
* Under the Hood
* Future Enhancements

Introduction:

This command line script reads from two files that have timestamped bandwidth values. It calculates the ratio of their bandwidths, the total number of bytes in each, as well as the ratio of the total number of bytes with user specified intervals. The user will be asked to enter this specified interval.

Input files: “R1.txt” and “R2.txt”

Output files: “R3.txt” and “R4.txt”

R3 holds the ratio of the bandwidths of R2 to R1.

R4 holds the ratio of the total bytes of R2 to R1 according to the user-specified interval.

Prerequisites for the Script to Work Properly:

1. Files must be named “R1.txt” and “R2.txt”
2. Division operations are always R2/R1
3. “R1.txt” and “R2.txt” should both have two columns which are tab separated on each line. First column should be time and second column should be bandwidth.
4. The first line of the file should be the first line of data. The file should not have column labels.
5. The first line of data should be the last time at which the bandwidth had a value of zero.
6. Bandwidth values can either be in normal format or in scientific notation.

Under the Hood:

Initially, the script will ask for the specific interval the user wants in order to generate the “R4.txt”. Then it reads from “R1.txt” and “R2.txt”, taking the ratios of the bandwidths (R2/R1) at each times along with multiplying first column of R1 (time) with second column of R1 (bandwidth) to get the total number of bytes.

If either R1 or R2 are have more lines of data than the other, the script disregards the extra data. For example, if R1 has 4 lines of data and R2 has 5 lines of data. The script will only take into account the first 4 lines of data in R2 because R1 only has 4 lines available. Since the first line of R1 and R2 are zeros, there will be a divide by zero error when computing the ratios. The script takes care of this with a simple conditional statement.

The ratios of the bandwidths from R2 and R1 are then written to a file called “R3.txt”. (Values are rounded to the nearest hundredths place.) Now the script prepares to calculate the ratio of the total bytes based on the interval specified by the user. For example, if the user wants the interval to be four, in R2, the script will take the fifth line of the total bytes and subtract it with the first line of total bytes. The same process occurs in R1 and the calculated value from R2 is divided by the calculated value from R1. The time recorded with this particular example will be the time from the fifth line of data. These values are then written to a file called “R4.txt”.

The script then closes every file and terminates.

Future Enhancements:

* Command line arguments (e.g. the files, the interval).
* Script should be easier to use (i.e. without prerequisites that are mentioned above).