**Project Report**

In this project we check if the internet usage of one user is statistically distinguishable when compared to the internet usage of the other users. We have performed this analysis by selecting 3 different time windows 10secs, 227 secs and 5 minutes. We compared the results for each time window in order to check how time window affects profiling.

P values of these 3 time windows for week 1 and 2 where one user is compared with every other user for all the 54 users is attached in the zip file as excel sheets(as this table is too large to fit in the word document).

From the values in the excel sheet we can see that for time window **10 seconds**, the number of user combinations that are indistinguishable from each other **(i.e, with p value > 0.05) is 1623** where the number of user combinations distinguishable **(i.e, P value <=0.05) is 1293**. I say user combinations because user 1’s week 1 is compared with user 2’s week 2, similarly user 2’s week 1 is compared with user 1’s week 2. If first combination is distinguishable does mean that second combination has to be distinguishable even if it is between the same set of users. Therefore, there are 54\*54 combinations of users with p value. Time taken to execute code for this time window is 5965 seconds.

For time window **227 seconds**, the number of user combinations that are indistinguishable from each other **(i.e, with p value > 0.05) is 1963,** where the number of user combinations distinguishable **(i.e, P value <=0.05) is 953**. Whereas time taken to execute code for this time window is 86 seconds.

For time window **5 minutes**, the number of user combinations that are indistinguishable from each other **(i.e, with p value > 0.05) is 2011** where the number of user combinations distinguishable **(i.e, P value <=0.05) is 905**. Whereas time taken to execute code for this time window is 76 seconds.

Hence, we can say that time window of 10 seconds is better in terms of authentication as the number of users who are statistically indistinguishable is very less in comparison with other windows.

I have considered the execution time to know the computational overhead each time window has. And from above execution times, we can see that time window of 10 seconds even though it produces lesser number of indistinguishable user combinations, its computational overhead and execution time is high.

When it come to the indistinguishability of the user with itself, we can see that the P values which are present in the diagonal of the table are ‘nan’, this is because r2a2a will always be 1 as the difference between their ranks is always 0(as it is week 2 of user 2 with week 2 of user 2). Because of this, Z value is always divided by 0, where z is calculated by 1-r2a2a in the denominator. Hence, we get value ‘nan’ when we calculate p value for user with himself.

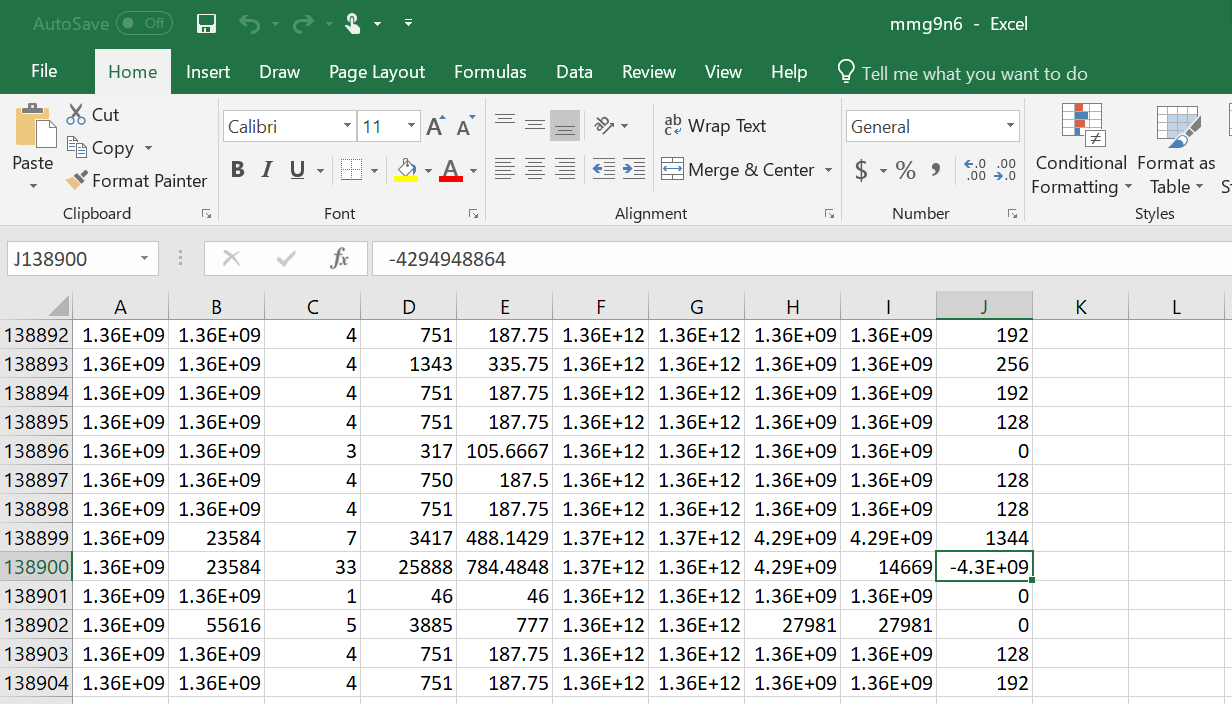
**Executing the Code:**

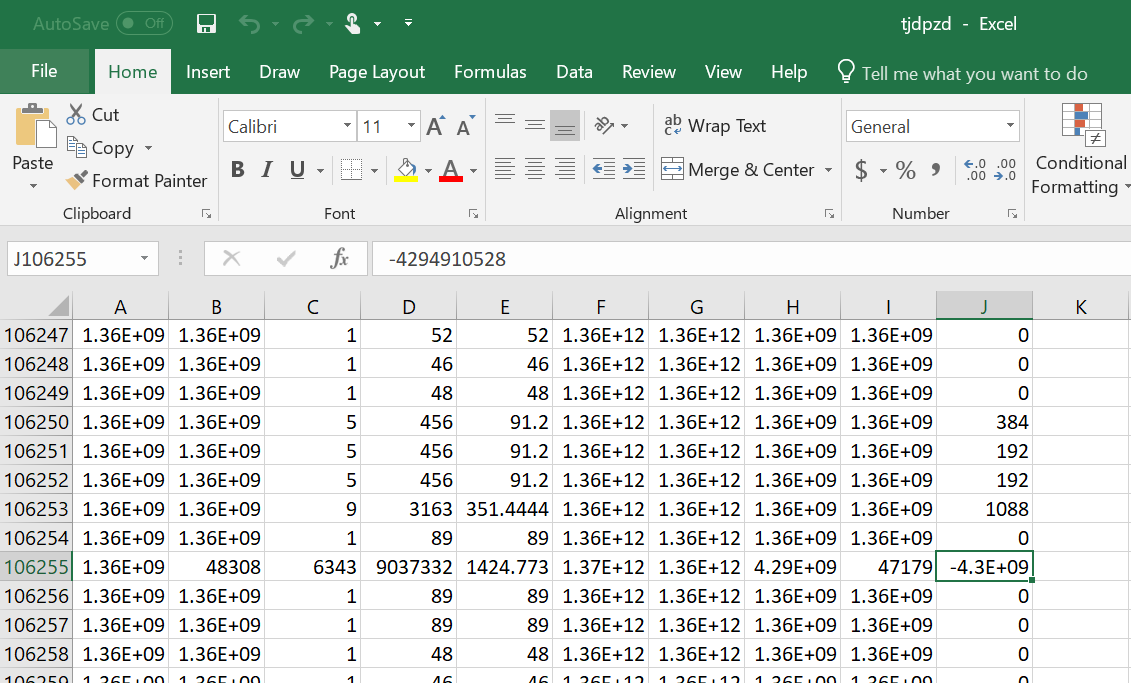
Download all the files from the zip folder attached into a folder where we have all the excel sheets for each user(convert all excel files to .csv comma delimited files). Check if there are a total of 57 files in your folder. 54 user excel files and 3 related to code and object files.

Open the file which has code and in line 39, mention the directory in which you have all these files with a “\\” between folders. Mention the time window for which you want to run the file in line 75. For example 10 , 227 , 300(5minutes). The value that is mentioned here should be in seconds. Hence, convert minute or hour time window to seconds.

Run the file which has code.

Note: I have changed 2 values in files mmg9n6 and tjdpzd because they have negative values in one duration field which is causing the program to stop at run time. Attached are the screen shots of the same. Duration can not be in negative hence changed the value to 0 to continue.





**Explanation of Code: (Procedure)**

Code is written in C++ Language.

Initially I have collected all files from a specific directory mentioned in the code and pushed these names into a files vector. As the vector size can dynamically change, we can have any number of files in the mentioned directory.

Opened a loop to read each file and perform some operations on the data. The file that is taken is a .csv file, hence the data is separate by a comma. We take advantage of this fact and collect the data of columns Doctets, FirstPacket and Duration into respective vectors. Rest of the columns data is skipped. Now, data in FirstPacket vector is converted to human readable form using below functions.

std::stringstream is(d);

is >> epoch64;

epoch64 /= 1000;

time\_t t = epoch64;

epo = ctime(&t);

The data we get from the above functions are in the form of “Wed Feb 06 20:01:57 2013”. This string is then tokenized and each piece of data is stored in its respective vector of Day, Month, Date, Hrs, Min, Sec, year. Using data in vectors Doctets and Duration, we calculate octets/duration for each record in the excel and store it in vector od. If Duration value is 0, the result gives a divide by zero error. Hence, these records are marked with a -1 to ignore them in further calculations.

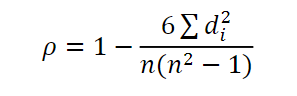
Assuming the day is divided into groups of time windows. We calculate the total number of seconds for each record in excel by multiplying Hrs with 24, Min with 60 and adding all these to seconds. This value is now divided with the length of the slot. This is done to categorize them into groups of a particular day.

The number of days is calculated by the Month value “Feb” we get when we convert the first packet to human readable form. This program works for any month. Assuming first record (apart from header) in every excel sheet gives the actual month information.

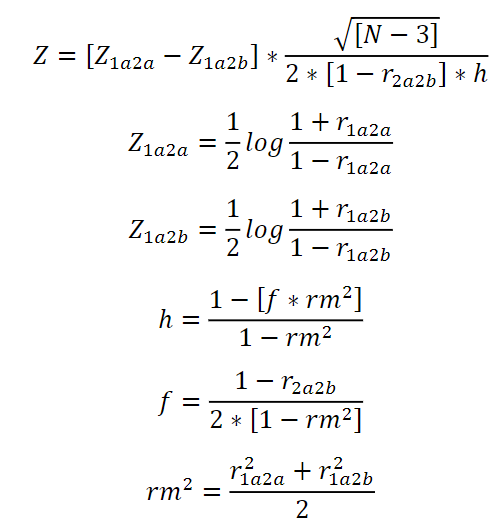
A matrix with rows number as Dates of the Month and columns number as groups of the day is created. Now for every row in excel sheet, we get its Date value and group value calculated before and add its doctets/duration value to the already present doctets/duration value in the matrix created above with Data as row value and group as y value. While doing this, the number of rows of excel sheet getting added in each cell is made a note of which is later used to get the average value of doctets/duration as a whole.

A matrix called filematrix is created with files as rows and all groups of every day in the month (except Saturday and Sunday) as columns. The Matrix starts with the first Monday of that month. Above calculated average doctets/duration value is pushed into this matrix for respective file in an order of days which is ordered with groups. Example, for[2,13] (if number of groups in a day is 10) the value in the cell is for 2nd file and 3rd group of the second day. I have added an additional condition of limiting to 10 days, as we are computing only for 1st and 2nd week. If this condition is removed, we can store the data for other weeks of the month in similar way.

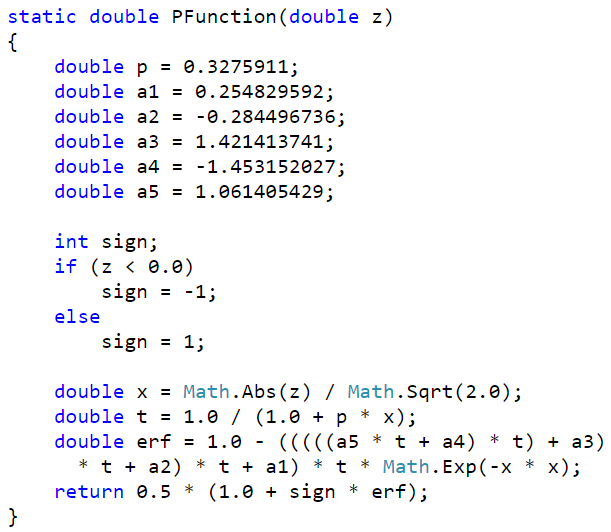
Using this filematrix, ranks are given to each group which has value of average doctets/durations in a particular day. For calculating Spearman’s correlation coefficient, the difference between the respective ranks (according to the r value) is calculated and applied in the below formula to get three correlation values r1a2a, r1a2b,r2a2b where numbers show weeks and characters are showing subjects.



These values r1a2a, r1a2b,r2a2b are used in below formulas to get Z value.



P values for these Z values are calculated using the below function.



When P≤0.05 means that correlation coefficient calculated for Internet usage patterns for an unknown subject (say b) is significantly smaller than that for a known subject (say a) and as such “subject b” will be identified as a subject distinct from “subject a”. On the contrary, when P> 0.05, indicates that correlation coefficient calculated for Internet usage patterns for an unknown subject (say b) is not significantly smaller than that for a known subject (say a), and as such “subject b” will be identified as indistinguishable from “subject a”.

The number of user combinations that are indistinguishable from each other (i.e, with p value > 0.05) and the number of user combinations distinguishable (i.e, P value <=0.05) is calculated.

The data in the output is copied into excel for better understandability.