Project: Profiling Internet Users
CIS6930 Information security and privacy

Ashutosh Modi (modia), MSCS
Department of Computer Science and Engineering
University of South Florida
Tampa, Florida 33620
Email: modia@mail.usf.edu

1. Objective

Internet usage of two users can be distinguishable or not. Observation needs to be done by changing time window.

2. Given Data

In this project data usage of 54 users are given in the form of excel file. This file contains following data columns:

- unix_secs: Current count of seconds since 0000 UTC 1970
- sysuptime: Current time in milliseconds since the export device booted
- dpkts: Packets in the flow - doctets: Total number of Layer 3 bytes in the packets of the flow
- doctets/dpkts: Packets in the flow - Real First Packet: Date and time in epoch
- Real End Packet first: Date and time in epoch
- Last: SysUptime at the time the last packet of the flow was received
- Duration: time in msec

3. Code

I have used C++ and MATLAB to perform the analysis. C++ was used to read the data and to convert it to usable format. Whereas, MATLAB was used for analysis.

Following are few assumptions made while performing data cleaning and analysis:

- Saturday and Sunday data have been excluded from the analysis part.
- Data having duration of 0 has been ignored from analysis.
- Excel file name has been considered as a unique user name.
- Zero value is used for no data in specific time slot

4. Analysis

To understand the distribution of the data histogram plot was used in MATLAB. Data can be visualized by plotting histogram as shown in figure 1 on the following page. This plot is for 9 users only for all 53 users histogram can be plotted as shown in figure 2 on the next page.

4.1. Algorithm

Pseudo code for implementation can be given as shown in pseudo code 1 on page 3 and pseudo code 2 on page 3. Algorithm 1 on page 3 are the main steps performed in data extraction process in C++. And algorithm 2 on page 3 are the main steps performed in data processing and analysis step in MATLAB. Sample data and slots created by C++ code are as shown in figure 3 on page 3 and figure 3b on page 3 respectively.

Duration zero case occurs when for some time slot there is not data observed.

4.2. OS and Tools

Following are the OS and tools details:

- Windows 10 (64 bit)
- Visual Studio 15
- MATLAB 2016a

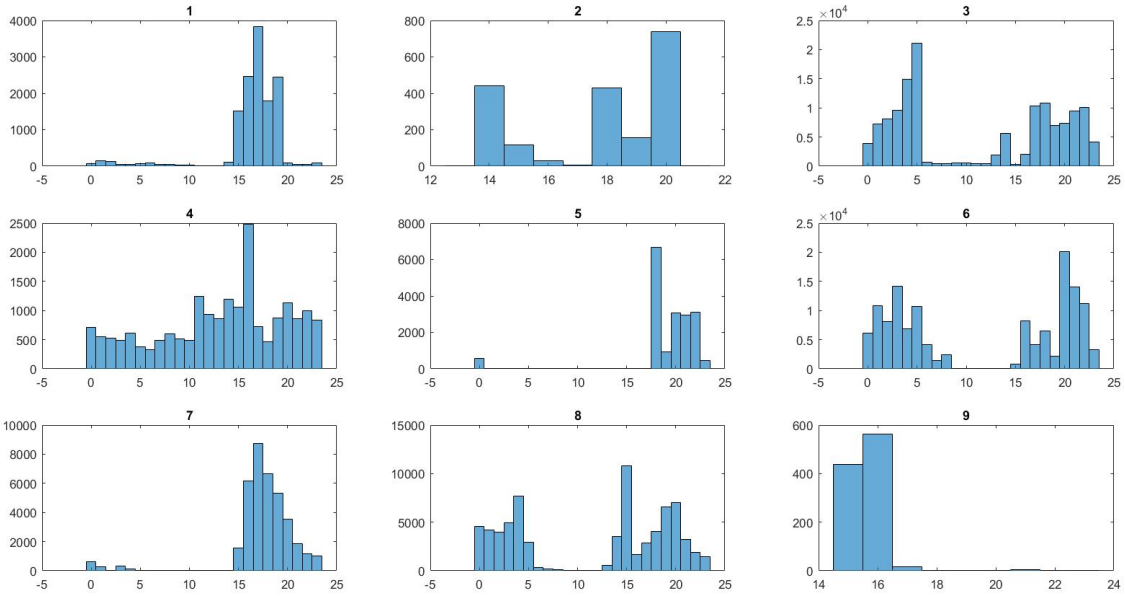


Figure 1: Histogram-1

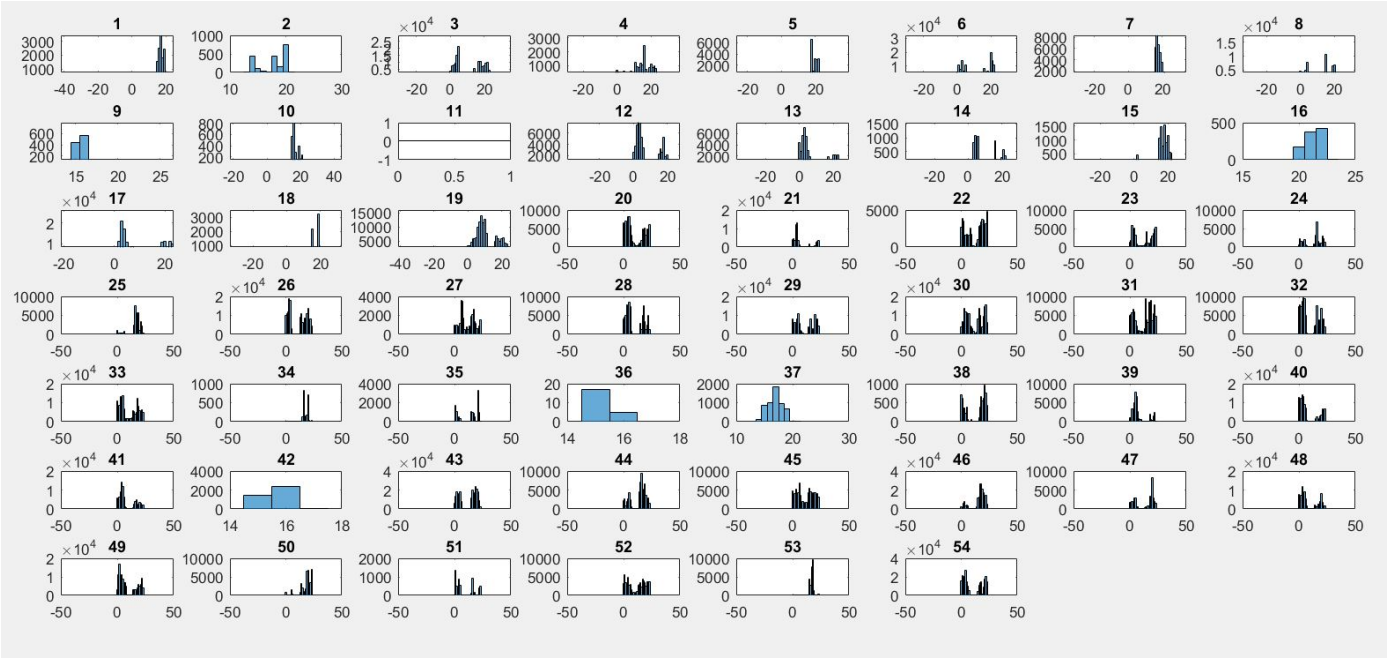


Figure 2: Histogram-2

5. How to execute code

C++ source needs to be compiled with required time window. Generated executable can be used to create data and slots stored as csv format. This can be used in matlab code to get the required pvalues and user count.

C++ Steps:

- Create a console project in visual studio for c++
- Use source.cpp as main code

Algorithm 1 C++

- File reading and error handling
- Creating time intervals; this creates desired time slots for 24 hrs. time
- Read all excel file names
  - Read each file line by line
  - Get required columns
  - Convert epoch time
  - Update required data in data structure
  - Close file
- Update time slot with 0 value if there is no entry found for that slot - Create .csv file to store time slots
- Create .csv file to store all consolidated data against file name used as unique identification.

	A	B	C	D	E
1	doctets	Duration	slot	Day	User
5088	4394	80960	767	4	ajb9b3.csv
5089	43079	67328	768	4	ajb9b3.csv
5090	70278	53312	769	4	ajb9b3.csv
5091	6647005	669632	770	4	ajb9b3.csv
5092	553298	393216	771	4	ajb9b3.csv
5093	174084	229632	772	4	ajb9b3.csv
5094	78032	166400	773	4	ajb9b3.csv
5095	158791	227904	774	4	ajb9b3.csv
5096	121788	170752	775	4	ajb9b3.csv
5097	71780	33536	776	4	ajb9b3.csv
5098	54562	70912	777	4	ajb9b3.csv
5099	30263	126272	778	4	ajb9b3.csv
5100	12333	139008	779	4	ajb9b3.csv
5101	15283	101440	780	4	ajb9b3.csv
5102	21330	83648	781	4	ajb9b3.csv
5103	105654	164288	782	4	ajb9b3.csv
5104	178428	163712	783	4	ajb9b3.csv
5105	30846	90496	784	4	ajb9b3.csv
5106	31750	130752	785	4	ajb9b3.csv
5107	20755	107520	786	4	ajb9b3.csv
5108	446427	178112	787	4	ajb9b3.csv
5109	3729077	202304	788	4	ajb9b3.csv
5110	443683	174144	789	4	ajb9b3.csv
5111	796322	135040	790	4	ajb9b3.csv

(a) User Data

	A	B
1	StartTime	EndTime
2	0:00:00	0:01:00
3	0:01:00	0:02:00
4	0:02:00	0:03:00
5	0:03:00	0:04:00
6	0:04:00	0:05:00
7	0:05:00	0:06:00
8	0:06:00	0:07:00
9	0:07:00	0:08:00
10	0:08:00	0:09:00
11	0:09:00	0:10:00
12	0:10:00	0:11:00
13	0:11:00	0:12:00
14	0:12:00	0:13:00
15	0:13:00	0:14:00
16	0:14:00	0:15:00
17	0:15:00	0:16:00
18	0:16:00	0:17:00
19	0:17:00	0:18:00
20	0:18:00	0:19:00
21	0:19:00	0:20:00
22	0:20:00	0:21:00
23	0:21:00	0:22:00
24	0:22:00	0:23:00
25	0:23:00	0:24:00

(b) Sample Slots

Figure 3: SampleData

Algorithm 2 MATLAB

- Read .csv files generated by C++ code
- Convert data to MATLAB data types
- For each unique user
  - Get user data in different tables to perform analysis
  - Calculate dOctets per duration
  - Handle case where in duration is zero
  - Consolidate data in table
- Concatenate data as per week of the month - Calculate “Spearman” correlation coefficient among different week data
- Check for the data wherein P value is greater than 0.05
- Count all such occurrences of combinations wherein p value is greater

- Build the source code in release mode in 64-bit option to create an executable
- Create new folder and copy this .exe file there.
- Copy all users' excel file at the same location where this exe is placed.
- Open command prompt using 'cmd' command
- Change directory to the desired folder using 'cd' command
- Run the executable in command prompt at this location
- Two files "Data.csv" and "Slots.csv" should get generated as same location

MATLAB Steps:

- Open 'Analysis.m' file in matlab
- Copy files "Data.csv" and "Slots.csv" to the same directory where 'Analysis.m' exists
- Run the code using 'Run' button in editor or use F5 to execute
- Result should get generated in Matches vector

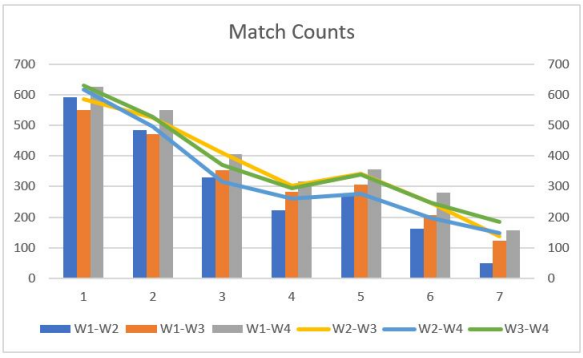
Need to perform steps mentioned above for different desired time slots.

6. Results

To check if two users are statistically distinguishable algorithm ran for different time window. Figure 4 shows the results obtained on these time window. Each value in columns represents Number of unique combinations of users such that their PValue is less than 0.05. for 'n' users, total combinations of users would be $n(n-1)/2$. In our case for 53 users this would be 1378. All these combinations need to be check across different weeks. For 4 weeks, W1, W2, W3, W4 of the month we can have 6 different combinations.

Match Count		Time slots in sec.						
		1800	900	300	180	227	120	60
Week Combinations	W1-W2	591	485	329	223	275	161	49
	W1-W3	549	471	354	282	307	206	124
	W1-W4	627	550	405	316	357	280	157
	W2-W3	586	525	410	303	342	247	136
	W2-W4	616	495	316	261	277	198	148
	W3-W4	629	529	370	295	338	247	184

(a) Table



(b) Chart

Figure 4: Result

7. Conclusion

In this analysis data was checked per week wise. For a month of data getting a trend and analyzing it may not give a clear trend. This is because for many users, data usage may not be consistent in one month. Data for few more month might give a promising trend to predict and analyze the behavior. Moreover, deciding a time window in a day is critical. In this report, as can it be seen in result chart, as time window made smaller and smaller less matches were found. But observed trend was not monotonous.