Profiling internet users CIS6930 Information Security and Privacy

Jaynab Khatun

UID- U06687764

Introduction:

The objective of this project is to find out at what time window a user is statically indistinguishable from the number of other user's data across. There are three time window 10s, 227s and 5 minutes and total 54 users.

Source of data:

The source of data for this project is Cisco NetFlow version 5. Many parameters has been provided like Packet, Octets, Real First Packet, Real End Packet, duration, Doctets/packet and many other variables. There are 54 data sheet for each user for a month long period.

Requirement:

1. Windows, 64 bit machine
2. Anaconda, Spider to run python3.6
3. Data for 54 users

Summary of contribution:

In this project, 54 user's data in excel sheet are collected, then modify it with parameter Doctet, Real First Packet, Duration. For ease to handle excel sheet other column has discarded as those are not needed. At same time, calculated the Doctet/Duration and ignore the duration value where it is zero to handle division zero. I have collected epoch data for 1st February, 8th February from internet which help to find time window calculation in program. Write program in python to divide the time window for 10s, 227 s and 5 minutes which define as 300s, the program name as first.py. While dividing time widow, count the frequency as well so that it will help to find out the total number window in week for particular user. In this way find out all data for week1, week2 and write excel sheet which named as New1 for 10s window, New2 for 227s window, New3 for 300s window.

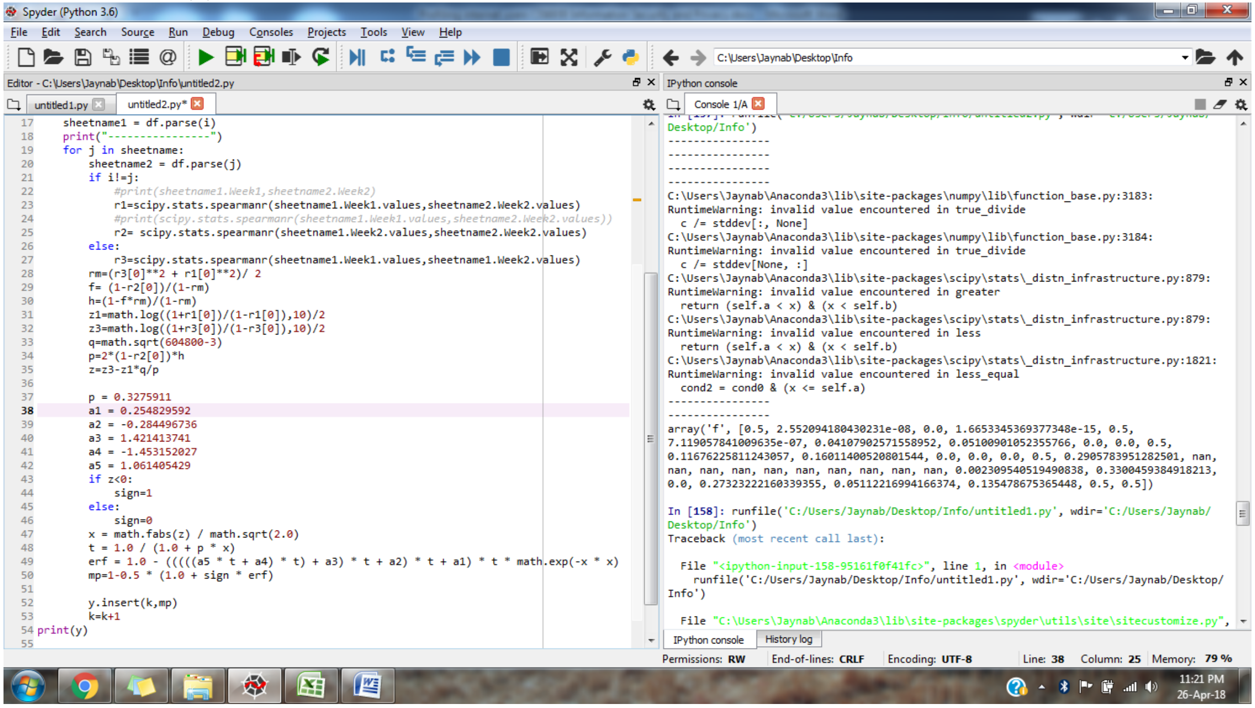
Now the second part of project is to find out Spearman Correlation coefficient and MRR-Test which is known as Z value,I have written another program in python named as second.py. I have used "spearmanr" function which is predefined in Python3.6 for each user and consider week1, week2. At the same time calculate the Z- value and P value for correlation coefficient of user and corresponding week. The collected all P value for each user and put it in a list of array. The image show

How to run in Python:

1. In 54 excel sheet, Doctet, Real First packet and Duration is selected and calculated the Doctet/duration. Name as one.xlsx, two.xlsx, three xlsx like this till 54 users.
2. Keep all modified excel sheet in same work folder where python program is created
3. Read all excel sheet and produce the new1, new2 and new3 excel sheet using python program.

4. Now take new1, new2, new3 excel sheet input and find the spearman correlation coefficient and corresponding Z-value, P-value using program second.py. The program is already written in python and read excel sheet. User needs to run to python program separately one by one(first run First.py the run second.py)

The below screen shot has taken for five user and print their corresponding P value using second.py



```python
    sheetname1 = df.parse(i)
    print("----------------")
    for j in sheetname:
        sheetname2 = df.parse(j)
        if i!=j:
            #print(sheetname1.Week1,sheetname2.Week2)
            r1=scipy.stats.spearmanr(sheetname1.Week1.values,sheetname2.Week2.values)
            #print(scipy.stats.spearmanr(sheetname1.Week1.values,sheetname2.Week2.values))
            r2= scipy.stats.spearmanr(sheetname1.Week2.values,sheetname2.Week2.values)
        else:
            r3=scipy.stats.spearmanr(sheetname1.Week1.values,sheetname1.Week2.values)
        rm=(r3[0]**2 + r1[0]**2)/ 2
        f= (1-r2[0])/(1-rm)
        h=(1-f*rm)/(1-rm)
        z1=math.log((1+r1[0])/(1-r1[0]),10)/2
        z3=math.log((1+r3[0])/(1-r3[0]),10)/2
        q=math.sqrt(604800-3)
        p=2*(1-r2[0])*h
        z=z3-z1*q/p

        p = 0.3275911
        a1 = 0.254829592
        a2 = -0.284496736
        a3 = 1.421413741
        a4 = -1.453152027
        a5 = 1.061405429
        if z<0:
            sign=1
        else:
            sign=0
        x = math.fabs(z) / math.sqrt(2.0)
        t = 1.0 / (1.0 + p * x)
        erf = 1.0 - (((((a5 * t + a4) * t) + a3) * t + a2) * t + a1) * t * math.exp(-x * x)
        mp=1-0.5 * (1.0 + sign * erf)

        y.insert(k,mp)
        k=k+1
print(y)
```

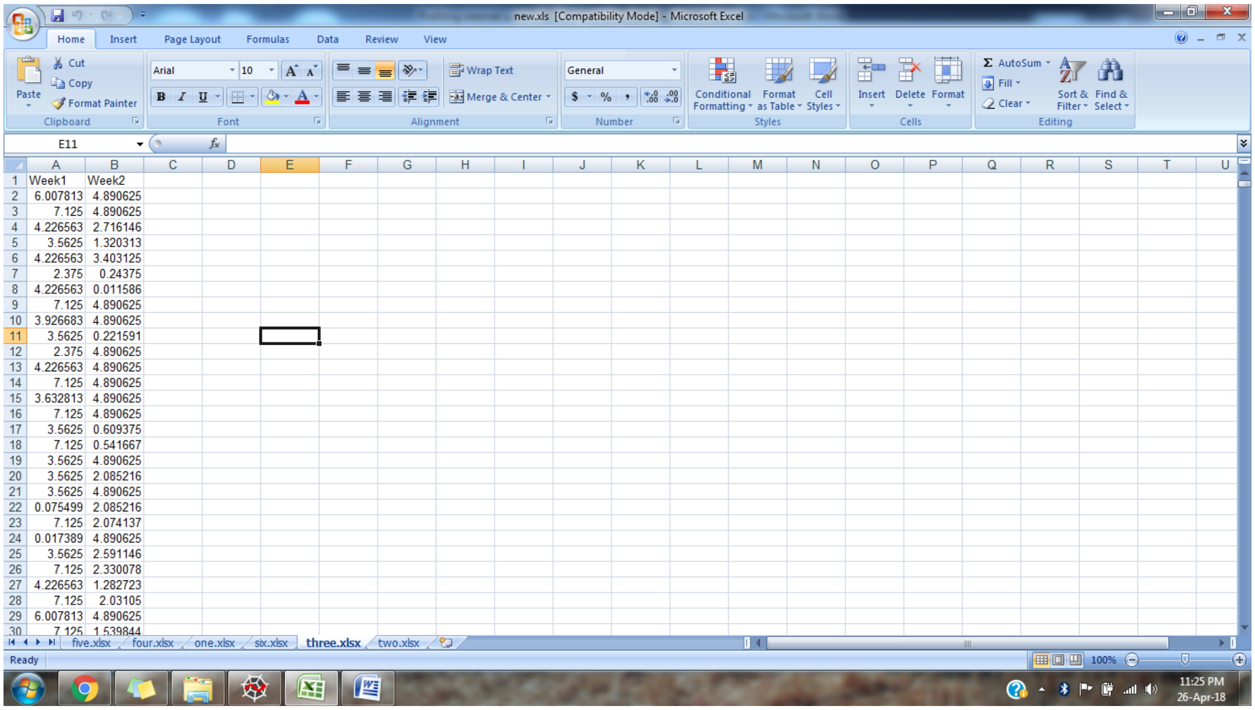
```
Desktop/Info')
----------------
----------------
----------------
----------------
C:\Users\Jaynab\Anaconda3\lib\site-packages\numpy\lib\function_base.py:3183:
RuntimeWarning: invalid value encountered in true_divide
  c /= stddev[:, None]
C:\Users\Jaynab\Anaconda3\lib\site-packages\numpy\lib\function_base.py:3184:
RuntimeWarning: invalid value encountered in true_divide
  c /= stddev[None, :]
C:\Users\Jaynab\Anaconda3\lib\site-packages\scipy\stats\_distn_infrastructure.py:879:
RuntimeWarning: invalid value encountered in greater
  return (self.a < x) & (x < self.b)
C:\Users\Jaynab\Anaconda3\lib\site-packages\scipy\stats\_distn_infrastructure.py:879:
RuntimeWarning: invalid value encountered in less
  return (self.a < x) & (x < self.b)
C:\Users\Jaynab\Anaconda3\lib\site-packages\scipy\stats\_distn_infrastructure.py:1821:
RuntimeWarning: invalid value encountered in less_equal
  cond2 = cond0 & (x <= self.a)
----------------
----------------
array('f', [0.5, 2.552094180430231e-08, 0.0, 1.6653345369377348e-15, 0.5,
7.119057841009635e-07, 0.04107902571558952, 0.05100901052355766, 0.0, 0.0, 0.5,
0.11676225811243057, 0.16011400520801544, 0.0, 0.0, 0.0, 0.5, 0.2905783951282501, nan,
nan, nan, nan, nan, nan, nan, nan, nan, nan, 0.002309540519490838, 0.3300459384918213,
0.0, 0.27323222160339355, 0.05112216994166374, 0.135478675365448, 0.5, 0.5])

In [158]: runfile('C:/Users/Jaynab/Desktop/Info/untitled1.py', wdir='C:/Users/Jaynab/
Desktop/Info')
Traceback (most recent call last):

  File "<ipython-input-158-95161f0f41fc>", line 1, in <module>
    runfile('C:/Users/Jaynab/Desktop/Info/untitled1.py', wdir='C:/Users/Jaynab/Desktop/
Info')

  File "C:\Users\Jaynab\Anaconda3\lib\site-packages\spyder\utils\site\sitecustomize.py",
```

Below screen shot is for New1 excel sheet created by executing the first.py program



	A	B
1	Week1	Week2
2	6.007813	4.890625
3	7.125	4.890625
4	4.226563	2.716146
5	3.5625	1.320313
6	4.226563	3.403125
7	2.375	0.24375
8	4.226563	0.011586
9	7.125	4.890625
10	3.926683	4.890625
11	3.5625	0.221591
12	2.375	4.890625
13	4.226563	4.890625
14	7.125	4.890625
15	3.632813	4.890625
16	7.125	4.890625
17	3.5625	0.609375
18	7.125	0.541667
19	3.5625	4.890625
20	3.5625	2.085216
21	3.5625	4.890625
22	0.075499	2.085216
23	7.125	2.074137
24	0.017389	4.890625
25	3.5625	2.591146
26	7.125	2.330078
27	4.226563	1.282723
28	7.125	2.03105
29	6.007813	4.890625
30	7.125	1.539844

Analysis:

Form the Table of P value, it is observe that for same user in same week and time window, the P value is 0.5. Look out for number of average P value in P value matrix for each time window and find the

10s window:

Week1 & Week2	One	Two	Three	Four	five
One	0.5	.0255209	0.0	0.166533 e-15	0.0711905
Two	0.5	0.5	0.0410790	0.0510	0.0
Three	0.2715938	0.1167	0.5	0.16011	0.0
Four	0.5	0.0290578	0.00230954	0.5	0.3300459
five	00	0.2732	0.511	0.5	0.5

Conclusion: The project has done in Python3.6 and result has shown in the table with P value for each user. Most of P values are greater that 0.05.

Reference:

[1] Soheil Sarmadi (Ph D Student of University of South Florida)

[2] Sriram Chellappan(Assistant professor of University of South Florida)