Final Q1 Analytics

April 14, 2020

Consider only the rows with country_id = "BDV" (there are 844 such rows). For each site_id, we can compute the number of unique user_id's found in these 844 rows. Which site_id has the largest number of unique users? And what's the number?

[3]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

[4]:
```
df=pd.read_csv("Adops & Data Scientist Sample Data - Q1 Analytics.csv")
```

[5]:
```
df
```

[5]:

	ts	user_id	country_id	site_id
0	2019-02-01 00:01:24	LC36FC	TL6	NOOTG
1	2019-02-01 00:10:19	LC39B6	TL6	NOOTG
2	2019-02-01 00:21:50	LC3500	TL6	NOOTG
3	2019-02-01 00:22:50	LC374F	TL6	NOOTG
4	2019-02-01 00:23:44	LCC1C3	TL6	QGO3G
…	…	…	…	…
3548	2019-02-07 23:56:57	LC3F13	TL6	QGO3G
3549	2019-02-07 23:58:36	LC3842	HVQ	3POLC
3550	2019-02-07 23:58:56	LC35EB	TL6	QGO3G
3551	2019-02-07 23:59:19	LC3842	HVQ	3POLC
3552	2019-02-07 23:59:37	LC3842	HVQ	3POLC

[3553 rows x 4 columns]

[6]:
```
BDV=df[df["country_id"]=="BDV"]
```

[16]:
```
BDV.groupby(['site_id']).nunique()
```

[16]:

site_id	ts	user_id	country_id	site_id
3POLC	5	2	1	1
5NPAU	716	544	1	1
NOOTG	122	90	1	1

[17]:
```
#groupby site id and count unique
#5NPAU and 544
BDV.groupby('site_id')['user_id'].nunique()
```

[17]:
```
site_id
3POLC      2
5NPAU    544
N0OTG     90
Name: user_id, dtype: int64
```

Between 2019-02-03 00:00:00 and 2019-02-04 23:59:59, there are four users who visited a certain site more than 10 times. Find these four users & which sites they (each) visited more than 10 times. (Simply provides four triples in the form (user_id, site_id, number of visits) in the box below.)

[18]:
```
#filter date
four=df[(df["ts"]>="2019-02-03 00:00:00")&(df["ts"]<="2019-02-04 23:59:59")]
```

[19]:
```
#groupby user and site id
sites=four.groupby(["user_id","site_id"]).count().reset_index()
```

[20]:
```
#find users who visited site more than 10 times by ts count
sites[sites['ts'] > 10]
```

[20]:

	user_id	site_id	ts	country_id
3	LC06C3	N0OTG	25	25
417	LC3A59	N0OTG	26	26
485	LC3C7E	3POLC	15	15
493	LC3C9D	N0OTG	17	17

For each site, compute the unique number of users whose last visit (found in the original data set) was to that site. For instance, user "LC3561"'s last visit is to "N0OTG" based on timestamp data. Based on this measure, what are top three sites? (hint: site "3POLC" is ranked at 5th with 28 users whose last visit in the data set was to 3POLC; simply provide three pairs in the form (site_id, number of users).)

[21]:
```
df
```

[21]:

	ts	user_id	country_id	site_id
0	2019-02-01 00:01:24	LC36FC	TL6	N0OTG
1	2019-02-01 00:10:19	LC39B6	TL6	N0OTG
2	2019-02-01 00:21:50	LC3500	TL6	N0OTG
3	2019-02-01 00:22:50	LC374F	TL6	N0OTG
4	2019-02-01 00:23:44	LCC1C3	TL6	QGO3G
…	…	…	…	…
3548	2019-02-07 23:56:57	LC3F13	TL6	QGO3G
3549	2019-02-07 23:58:36	LC3842	HVQ	3POLC
3550	2019-02-07 23:58:56	LC35EB	TL6	QGO3G
3551	2019-02-07 23:59:19	LC3842	HVQ	3POLC

```
3552  2019-02-07 23:59:37  LC3842        HVQ   3POLC

[3553 rows x 4 columns]
```

[22]:
```
#find last visit
df_last=df.sort_values('ts').groupby('user_id').tail(1)
df_last.groupby("site_id").nunique()
```

[22]:

site_id	ts	user_id	country_id	site_id
3POLC	28	28	5	1
5NPAU	990	992	3	1
EUZ/Q	1	1	1	1
GVOFK	42	42	1	1
JSUUP	1	1	1	1
NOOTG	561	561	6	1
QGO3G	288	289	1	1
RT9Z6	2	2	1	1

[23]:
```
#top three sites are 5NPAU, NOOTG, OGO3G
```

[24]:
```
df_last.groupby("site_id").nunique()
```

[24]:

site_id	ts	user_id	country_id	site_id
3POLC	28	28	5	1
5NPAU	990	992	3	1
EUZ/Q	1	1	1	1
GVOFK	42	42	1	1
JSUUP	1	1	1	1
NOOTG	561	561	6	1
QGO3G	288	289	1	1
RT9Z6	2	2	1	1

For each user, determine the first site he/she visited and the last site he/she visited based on the timestamp data. Compute the number of users whose first/last visits are to the same website. What is the number?

1 Considering users that have more than 1 time stamp

[38]:
```
df[df["user_id"]=="LC39B6"]
```

[38]:
```
                   ts user_id country_id site_id
1  2019-02-01 00:10:19  LC39B6        TL6   NOOTG
```

[56]:
```
count2=df.groupby("user_id").count()['ts'].reset_index()
filtered=count2[count2['ts']>1]["user_id"]
twotimestamps=df[df['user_id'].isin(list(filtered))]
```

[61]:
```
two_time_first=twotimestamps.sort_values('ts').groupby('user_id').head(1)
df_first
```

[61]:
```
                    ts user_id country_id site_id
0  2019-02-01 00:01:24  LC36FC        TL6   NOOTG
```

[62]:
```
two_time_last=twotimestamps.sort_values('ts').groupby('user_id').tail(1)
two_time_last
```

[62]:
```
                     ts user_id country_id site_id
28    2019-02-01 02:36:51  LC3DAD        TL6   NOOTG
39    2019-02-01 03:16:24  LC37EA        TL6   QGO3G
68    2019-02-01 04:45:22  LC305A        TL6   3POLC
150   2019-02-01 11:51:08  LC3222        TL6   QGO3G
163   2019-02-01 12:06:39  LC8C37        QLT   5NPAU
…                     …      …         …      …
3538  2019-02-07 23:14:19  LC3780        TL6   QGO3G
3540  2019-02-07 23:30:43  LC33F8        TL6   QGO3G
3545  2019-02-07 23:44:34  LC3561        TL6   NOOTG
3547  2019-02-07 23:55:07  LC3837        TL6   RT9Z6
3552  2019-02-07 23:59:37  LC3842        HVQ   3POLC

[655 rows x 4 columns]
```

[63]:
```
join_two=pd.merge(two_time_first, two_time_last, on='user_id')
join_two
```

[63]:
```
                    ts_x user_id country_id_x site_id_x                ts_y  \
0    2019-02-01 00:01:24  LC36FC          TL6     NOOTG  2019-02-07 00:24:50
1    2019-02-01 00:22:50  LC374F          TL6     NOOTG  2019-02-03 04:50:43
2    2019-02-01 00:24:21  LC3E1D          HVQ     GVOFK  2019-02-04 12:26:52
3    2019-02-01 00:25:29  LC3561          TL6     3POLC  2019-02-07 23:44:34
4    2019-02-01 00:29:15  LC3A01          TL6     NOOTG  2019-02-02 18:06:55
..                    …      …           …       …                  …
650  2019-02-07 19:05:16  LC342F          QLT     5NPAU  2019-02-07 19:13:48
651  2019-02-07 19:14:14  LCC3D7          QLT     5NPAU  2019-02-07 19:23:01
652  2019-02-07 19:23:29  LC362E          QLT     5NPAU  2019-02-07 19:30:22
653  2019-02-07 19:36:05  LC3D07          TL6     NOOTG  2019-02-07 21:42:23
654  2019-02-07 19:38:55  LC3557          BDV     5NPAU  2019-02-07 22:19:25

    country_id_y site_id_y
0            TL6     NOOTG
1            TL6     NOOTG
```

```
2            QLT     5NPAU
3            TL6     NOOTG
4            TL6     NOOTG
..           …       …
650          QLT     5NPAU
651          QLT     5NPAU
652          QLT     5NPAU
653          BDV     5NPAU
654          QLT     5NPAU

[655 rows x 7 columns]
```

[66]:
```
sum(join_two["site_id_x"]!=join_two["site_id_y"])
```

[66]:
```
246
```

[65]:
```
sum(join_two["site_id_x"]==join_two["site_id_y"])
```

[65]:
```
409
```

2 Considering All Users

[25]:
```
#first visits
df_first=df.sort_values('ts').groupby('user_id').head(1)
df_first
```

[25]:
```
                      ts user_id country_id site_id
0     2019-02-01 00:01:24  LC36FC        TL6   NOOTG
1     2019-02-01 00:10:19  LC39B6        TL6   NOOTG
2     2019-02-01 00:21:50  LC3500        TL6   NOOTG
3     2019-02-01 00:22:50  LC374F        TL6   NOOTG
4     2019-02-01 00:23:44  LCC1C3        TL6   QGO3G
…                    …      …         …      …
3542  2019-02-07 23:39:33  LC34C6        HVQ   GVOFK
3543  2019-02-07 23:41:25  LCC36A        TL6   NOOTG
3544  2019-02-07 23:42:35  LC34B8        TL6   QGO3G
3548  2019-02-07 23:56:57  LC3F13        TL6   QGO3G
3550  2019-02-07 23:58:56  LC35EB        TL6   QGO3G

[1916 rows x 4 columns]
```

[26]:
```
#last visit
df_last
```

[26]:
```
                     ts user_id country_id site_id
1     2019-02-01 00:10:19  LC39B6        TL6   NOOTG
2     2019-02-01 00:21:50  LC3500        TL6   NOOTG
4     2019-02-01 00:23:44  LCC1C3        TL6   QGO3G
11    2019-02-01 00:41:50  LCC3C3        QLT   5NPAU
12    2019-02-01 00:42:13  LC39C8        TL6   QGO3G
…                     …     …         …     …
3545  2019-02-07 23:44:34  LC3561        TL6   NOOTG
3547  2019-02-07 23:55:07  LC3837        TL6   RT9Z6
3548  2019-02-07 23:56:57  LC3F13        TL6   QGO3G
3550  2019-02-07 23:58:56  LC35EB        TL6   QGO3G
3552  2019-02-07 23:59:37  LC3842        HVQ   3POLC

[1916 rows x 4 columns]
```

[27]:
```
#join
join=pd.merge(df_last, df_first, on='user_id')
join
```

[27]:
```
                     ts_x user_id country_id_x site_id_x                 ts_y  \
0     2019-02-01 00:10:19  LC39B6          TL6     NOOTG  2019-02-01 00:10:19
1     2019-02-01 00:21:50  LC3500          TL6     NOOTG  2019-02-01 00:21:50
2     2019-02-01 00:23:44  LCC1C3          TL6     QGO3G  2019-02-01 00:23:44
3     2019-02-01 00:41:50  LCC3C3          QLT     5NPAU  2019-02-01 00:41:50
4     2019-02-01 00:42:13  LC39C8          TL6     QGO3G  2019-02-01 00:42:13
…                     …     …           …       …                   …
1911  2019-02-07 23:44:34  LC3561          TL6     NOOTG  2019-02-01 00:25:29
1912  2019-02-07 23:55:07  LC3837          TL6     RT9Z6  2019-02-03 03:30:25
1913  2019-02-07 23:56:57  LC3F13          TL6     QGO3G  2019-02-07 23:56:57
1914  2019-02-07 23:58:56  LC35EB          TL6     QGO3G  2019-02-07 23:58:56
1915  2019-02-07 23:59:37  LC3842          HVQ     3POLC  2019-02-05 16:21:30

     country_id_y site_id_y
0             TL6     NOOTG
1             TL6     NOOTG
2             TL6     QGO3G
3             QLT     5NPAU
4             TL6     QGO3G
…           …       …
1911          TL6     3POLC
1912          QLT     5NPAU
1913          TL6     QGO3G
1914          TL6     QGO3G
1915          HVQ     3POLC

[1916 rows x 7 columns]
```

[28]:
```
sum(join["site_id_x"]!=join["site_id_y"])
```

[28]: 246

[32]:
```
sum(join["site_id_x"]==join["site_id_y"])
```

[32]: 1670

[29]:
```
test = join.drop_duplicates(subset=['user_id'], keep=False)
```

[33]:
```
test
```

[33]:

	ts_x	user_id	country_id_x	site_id_x	ts_y	\
0	2019-02-01 00:10:19	LC39B6	TL6	NOOTG	2019-02-01 00:10:19	
1	2019-02-01 00:21:50	LC3500	TL6	NOOTG	2019-02-01 00:21:50	
2	2019-02-01 00:23:44	LCC1C3	TL6	QGO3G	2019-02-01 00:23:44	
3	2019-02-01 00:41:50	LCC3C3	QLT	5NPAU	2019-02-01 00:41:50	
4	2019-02-01 00:42:13	LC39C8	TL6	QGO3G	2019-02-01 00:42:13	
…	…	…	…	…	…	
1911	2019-02-07 23:44:34	LC3561	TL6	NOOTG	2019-02-01 00:25:29	
1912	2019-02-07 23:55:07	LC3837	TL6	RT9Z6	2019-02-03 03:30:25	
1913	2019-02-07 23:56:57	LC3F13	TL6	QGO3G	2019-02-07 23:56:57	
1914	2019-02-07 23:58:56	LC35EB	TL6	QGO3G	2019-02-07 23:58:56	
1915	2019-02-07 23:59:37	LC3842	HVQ	3POLC	2019-02-05 16:21:30	

	country_id_y	site_id_y
0	TL6	NOOTG
1	TL6	NOOTG
2	TL6	QGO3G
3	QLT	5NPAU
4	TL6	QGO3G
…	…	…
1911	TL6	3POLC
1912	QLT	5NPAU
1913	TL6	QGO3G
1914	TL6	QGO3G
1915	HVQ	3POLC

[1916 rows x 7 columns]

[ ]:
```
##1670 users went to same first and last website
```