bicincittaProblems

May 6, 2015

Contents

0.1 Problems in the Bicincitta data set from 2013

There are problems with the Bicincitta data that we need to address before loading the data into a reliable and proper data-base. We will point out these problems using examples, and measure their magnitude using systematic analysis, and then speculate about the source of these problems.

0.1.1 Data

We will load the data from JSONs provided to us by Bicincitta at the end of April 2015.

The resulting data is in the form of lists of dictionaries. We can take a peek at the keys in each of the four data types, by creating a data frame and displaying the first few rows.

```
a subnetwork is described by,
        id
        name

 a station is described by,
        name
        longitude
        subnetwork_name
        latitude
        id
        subnetwork_id

 a user is described by
        subnetwork_id
        gender
        expires
        postal_code
        subnetwork_name
        address
        id

 a transaction is described by,
        direction
        user_id
        station_id
        event_time
        id
```

The resulting dictionaries have ids that are UTF-8 strings. We can change these to integers to make our work easier,

There are keys in a transaction that do not seem to correspond to the data, but refer to the time at which the data was loaded into the JSON provided to us. We will drop these variables, and change the *event_time* to a time object. Following that we will sort the transactions by the event time

We also sort the subnetworks and stations by their *id*.

Stations and users have been assigned a *subnetwork_id* in the data. We can add the subnetwork name to these data,

0.1.2 Who are the users?

The simplest question may be the fraction of females vs males,

```
Of all the users  60  percent are female  and  39  percent males.
```

It would be interesting if 60% of the users were in fact female. However, as we will see later there seems to be a problem of user duplicacy biased towards females.

0.1.3 Subnetworks for stations and users

Are the *subnetwork_id*s for stations and users sensible? The subnetworks that the stations fall in are

```
Subnetworks that have stations assigned to them

          id               name
     0     1            La Cote
     1     2     Agglo Fribourg
     2     3              Bulle
     3     4    Les Lacs-Romont
     4     6           Chablais
     5     7     Valais Central
     6     8  Yverdon-les-Bains
     7     9    Lausanne-Morges
     8    10             Campus
     9    11            Riviera
     10   12    Lugano-Paradiso
```

We see that stations cover only 11 of the 18 subnetworks. Looking at the subnetworks with no stations,

```
Subnetworks without any assigned stations

         id       name
     0    5       Bâle
     1   13  PubliBike
     2   14      Vevey
     3   15     Morges
     4   16      Ouchy
     5   17   Paradiso
     6   18       Cern
```

we can see why there are no stations corresponding to these subnetworks. Basel, and Cern because Bicincitta have not given us data for these regions. Vevey, Morges, Ouchy, and Paradiso include stations subsumed by other subnetworks. These networks may be a remnant from previous versions of the data. This leaves **PubliBike** unexplained. As it turns out, there are users that have been assigned the subnetwork PubliBike (*id* 13). In fact we see later that the users who have registered transactions in the data have been assigned only PubliBike, and no other subnetwork.

```
Number of users from subnetwork PubliBike 58927
```

Users that have been assigned subnetwork PubliBike compose 70% of the total users in the data. However we are not going to see PubliBike in any of their transactions as there are no stations for the subnetwork PubliBike! What about other subnetworks without stations?

```
subnetworks assigned to users
```

```
Out[607]:     id               name
          0    2     Agglo Fribourg
          1    6           Chablais
          2    7     Valais Central
          3    8  Yverdon-les-Bains
          4    9    Lausanne-Morges
          5   10             Campus
          6   12    Lugano-Paradiso
          7   13          PubliBike
          8   14              Vevey
          9   16              Ouchy
          10  17           Paradiso
          11  18               Cern
```

```
subnets without any users
```

```
Out[608]:    id              name
          0   1           La Cote
          1   3             Bulle
          2   4  Les Lacs-Romont
          3   5              Bâle
          4  11           Riviera
          5  15            Morges
```

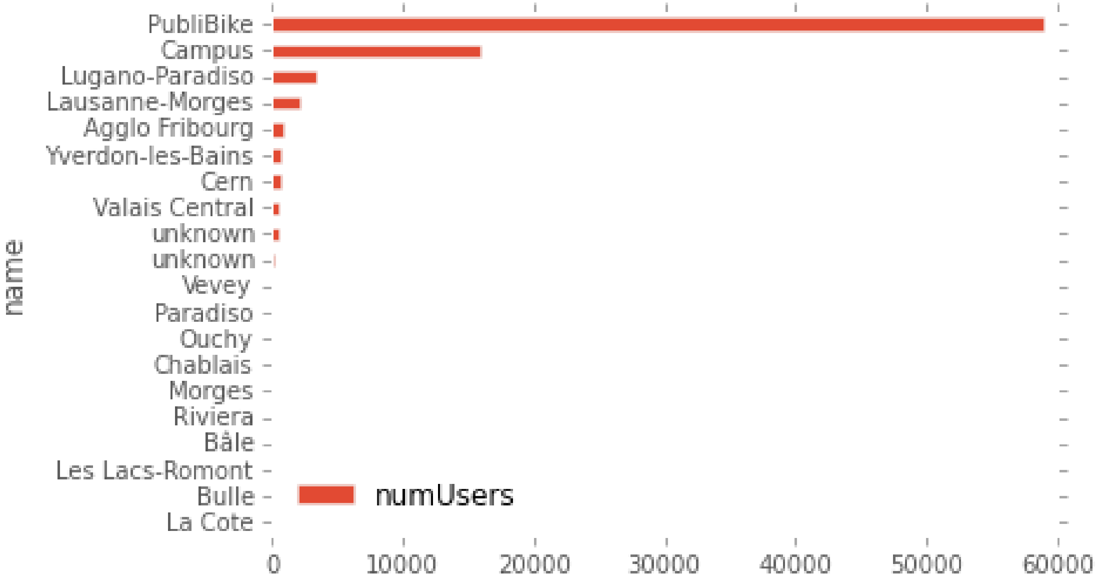
Comparing the subnets for users to subnets with stations we see that there are **only 7 subnets** for which we have stations as well as users,

```
subnets with stations as well as users
```

```
Out[609]:    id               name
          0   2     Agglo Fribourg
          1   6           Chablais
          2   7     Valais Central
          3   8  Yverdon-les-Bains
          4   9    Lausanne-Morges
          5  10             Campus
          6  12    Lugano-Paradiso
```

As a summary, let us tabulate the fraction of users in each of the subnets,

```
Out[611]:       id               name  numUsers
         0      1            La Cote         0
         1      3              Bulle         0
         2      4    Les Lacs-Romont         0
         3      5               Bâle         0
         4     11            Riviera         0
         5     15             Morges         0
         6      6           Chablais         3
         7     16              Ouchy         3
         8     17           Paradiso         3
         9     14              Vevey         4
         10  2011            unknown       152
         11  2005            unknown       469
         12     7     Valais Central       530
         13    18               Cern       721
         14     8  Yverdon-les-Bains       773
         15     2     Agglo Fribourg       810
         16     9    Lausanne-Morges      2183
         17    12    Lugano-Paradiso      3425
         18    10             Campus     15871
         19    13          PubliBike     58927
```



So, most of the users are in PubliBike, which could create a problem as there are no stations associated to PubliBike. Lets first look at the transactions before we try to find a solution to this problem.

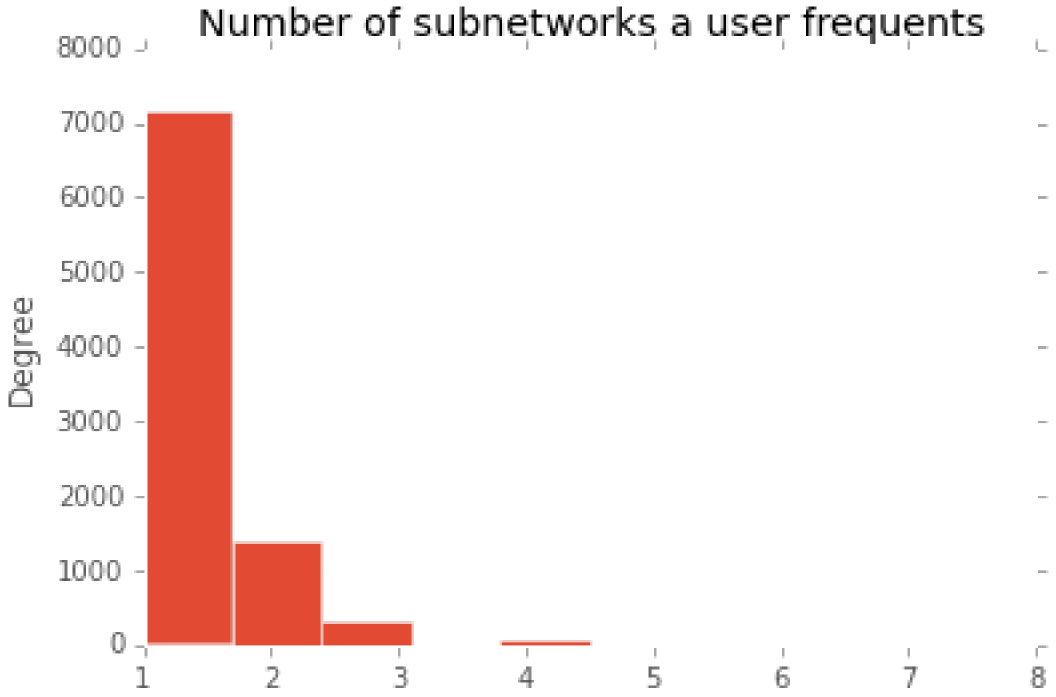
0.1.4 Transaction users, stations, and subnetworks

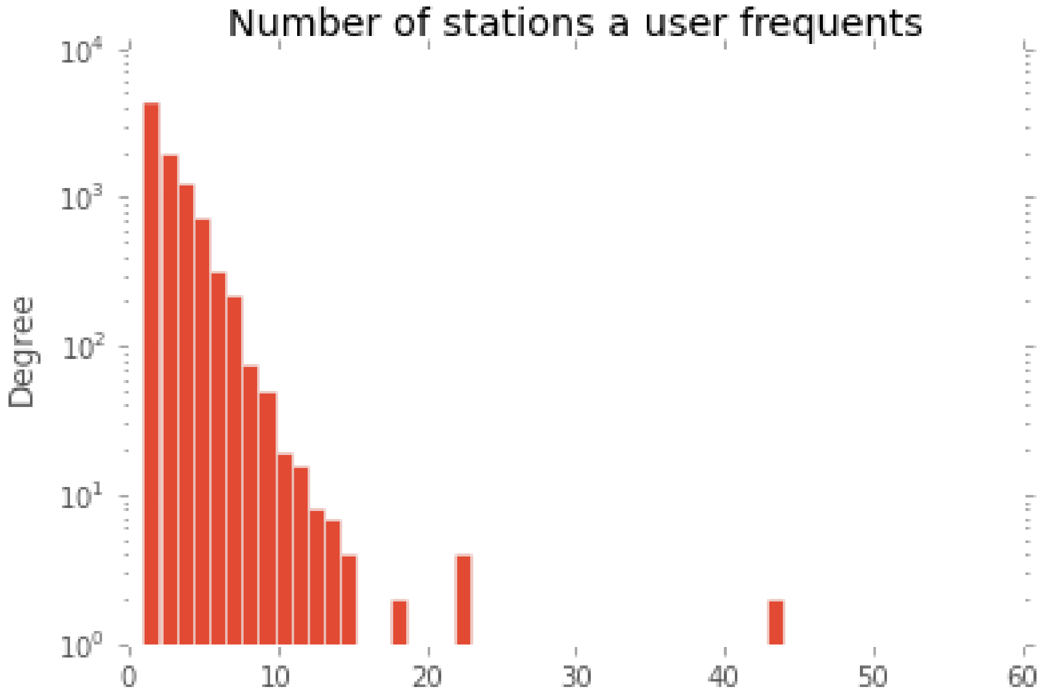
We begin by looking at how many users actually use the bike system (have valid transactions)

```
fraction of users who have registered a transaction 0.10673152586
```

Only 10% users have registered a transaction! Are all the transactions of a user in the same subnetwork?

```
Out[800]:       user_id  assigned_subnet  numStations  numSubnets  numTrxns
          3462   108132               13           56           8      2180
          4469   111523               13           46           7      2110
          5399    84545               13           43           7      5134
          3463   108133               13           43           6      3280
          4673   112287               13           23           3       540
```


Number of subnetworks a user frequents


Number of stations a user frequents

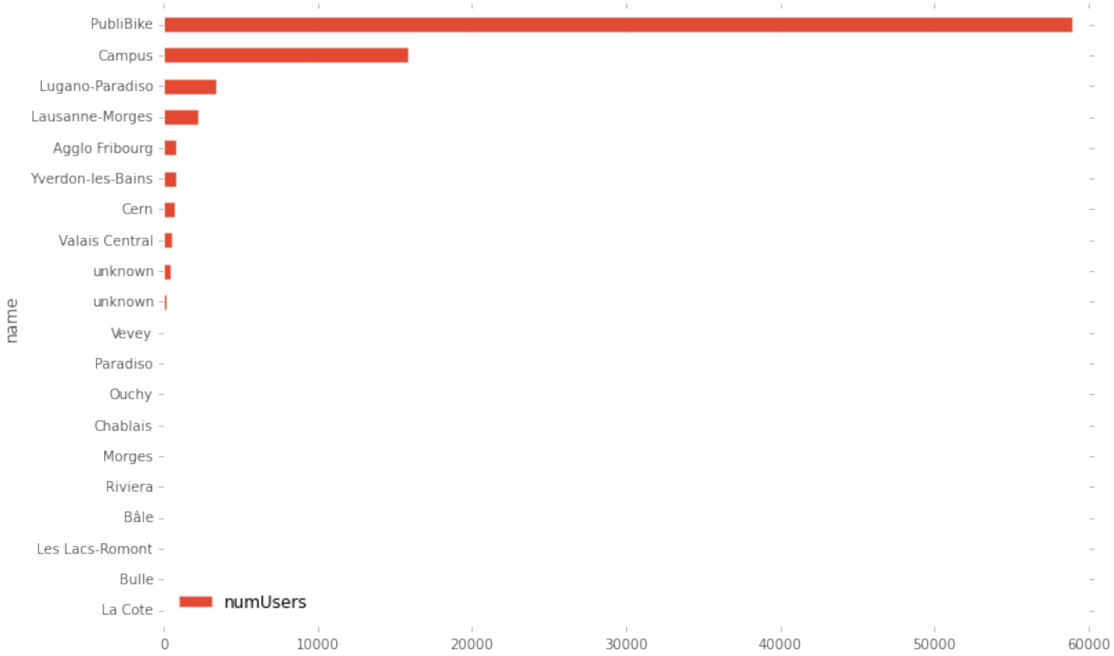
A simple question might guide us. How many transactions belong to Publibike users (who have subnetwork_id PubliBike) ?

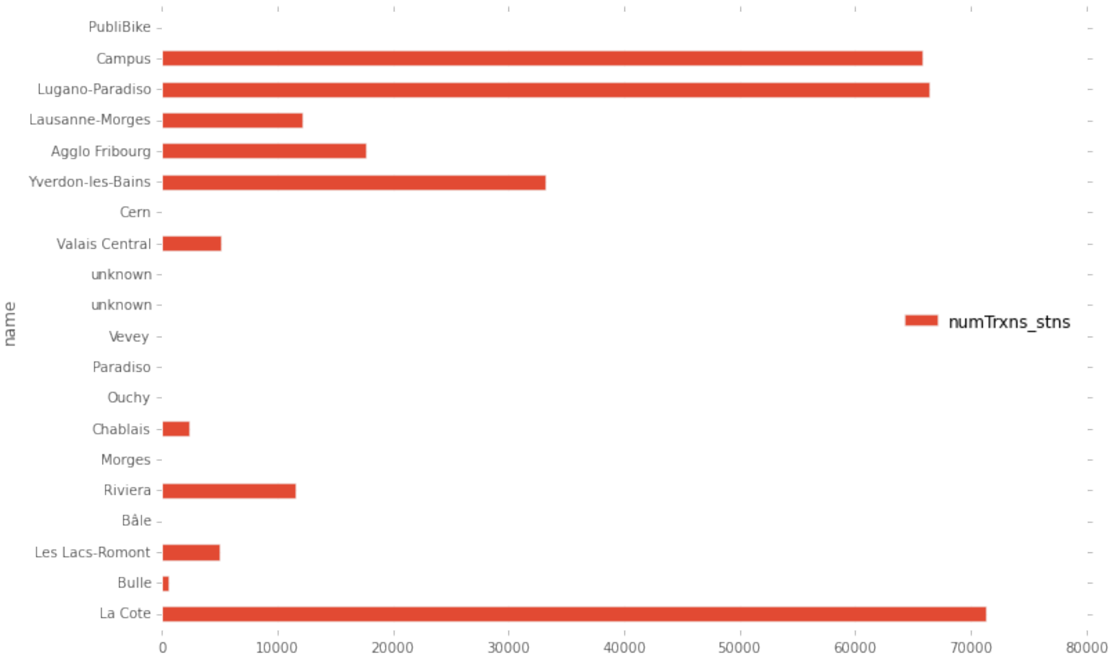
```
The subnetworks of the users who make transactions make a set([13])
```

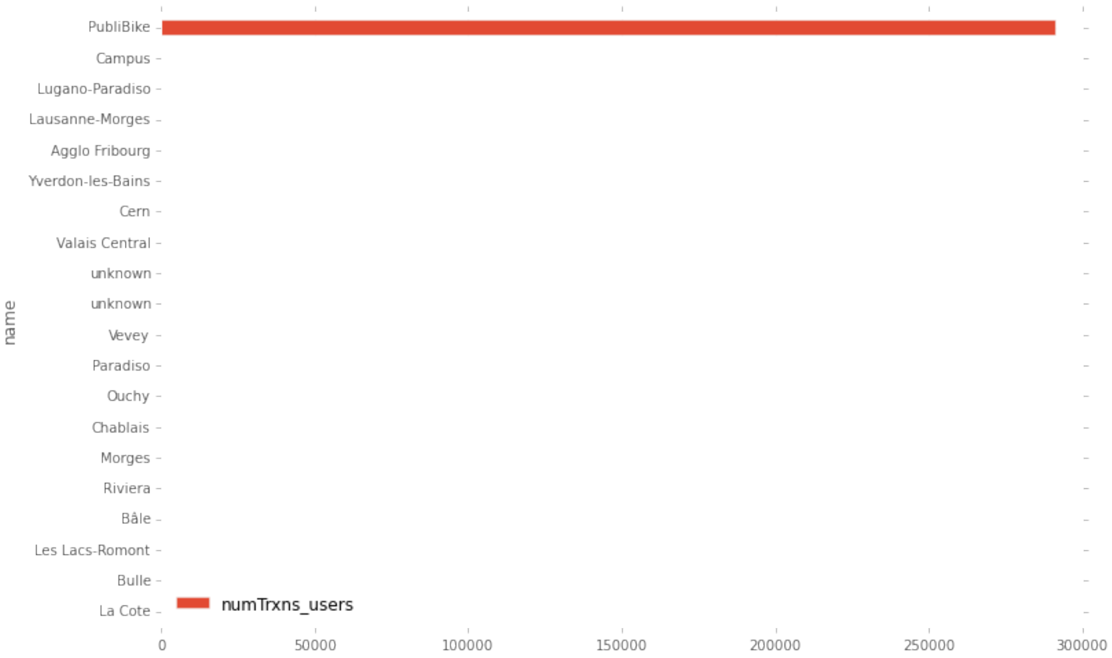
So, **from the view point of the users, all the transactions are in the subnetwork PubliBike**

How many transactions in a subnetwork of the stations?

Now we can make a table for subnetworks, counting the number of trxns through user and station *subnetwork_id*, in addition to the number of users.







```
Out[764]:                  id          name  numStations  numTrxns_stns  \
          name
          Vevey            14         Vevey            0              0
          Cern             18          Cern            0              0
          unknown        2005       unknown            0              0
          unknown        2011       unknown            0              0
```

```
Paradiso             17          Paradiso            0              0
Ouchy                16             Ouchy            0              0
PubliBike            13         PubliBike            0              0
Morges               15            Morges            0              0
Bâle                  5              Bâle            0              0
Bulle                 3             Bulle            2            566
Riviera              11           Riviera            5          11576
Valais Central        7    Valais Central            7           5077
Les Lacs-Romont       4   Les Lacs-Romont            9           4964
Yverdon-les-Bains     8 Yverdon-les-Bains            9          33227
Agglo Fribourg        2    Agglo Fribourg           10          17630
Chablais              6          Chablais           10           2377
Lausanne-Morges       9   Lausanne-Morges           11          12157
Lugano-Paradiso      12   Lugano-Paradiso           13          66415
La Cote               1           La Cote           13          71292
Campus               10            Campus           15          65853

                   numTrxns_users  numUsers
name
Vevey                           0         4
Cern                            0       721
unknown                         0       469
unknown                         0       152
Paradiso                        0         3
Ouchy                           0         3
PubliBike                  291134     58927
Morges                          0         0
Bâle                            0         0
Bulle                           0         0
Riviera                         0         0
Valais Central                  0       530
Les Lacs-Romont                 0         0
Yverdon-les-Bains               0       773
Agglo Fribourg                  0       810
Chablais                        0         3
Lausanne-Morges                 0      2183
Lugano-Paradiso                 0      3425
La Cote                         0         0
Campus                          0     15871
```

0.1.5 User addresses

There are several problems associated with user addresses. We have already noticed, and fixed, that the provided addresses in the JSON have not been *unquoted* from their web encoding. Here we continue to explore other problems that may arise in the addresses.

We want to count the number of users at one address. Because the addresses have been provided as strings, we have to be able to aggregate all address strings that describe the same address. We have written a python function to do this task, which takes the address and postal-code strings to provide a combined string taking into account some empirical disambiguation criteria such as *Av, Ave*, for *Avenue*.

```
number of users with available address 21659
number of these addresses that are unique 15446
```

What fraction of unique addresses have multiple users?

```
0.230545124951
```

How many users at addresses with multiple users?

```
9774
```

which corresponds to a fraction of all users with available address,

```
0.451267371531
```

Multiple users at the same address could be actual multiple people, or multiple registrations by the same person, or a glitch in the data. We can consider as an example the address with the most multiplicity of 53,

```
address:  via lambertenghi 1; 6900 , number of users:  53
```

We could say more about the multiple users at the same address if we look at their transactions. However as it turns out, we **do not have addresses for users who have registered transactions in the data**,

```
False
```

We can look at the subnetwork with addresses assigned to the *multi* users,

```
subnetworks with addresses assigned to multiple users set([2, 7, 8, 9, 10, 12, 18, 2005, 2011])
```

Some of the multi-user addresses have more than one subnetworks (through the users at that address)

```
Subnetworks for users living at addresses with multiple registered users
```

There are as many as 37 users assigned to the same address that also have more than subnetwork assigned. Addresses with several users might represent problems of multiple subscription. For example, if we look at addresses with more than 10 users,

```
Out[806]:                        address  numFemales  numMales  numUsers  \
          143   avenue des bains 9; 1007          37         0        37
          40    route cantonale 33; 1025          25         0        25
          105  avenue des bains 11; 1007          23         0        23
          48    place du tunnel 17; 1005          23         0        23

                                subnetworks
          143  set([Lausanne-Morges, Campus])
          40   set([Lausanne-Morges, Campus])
          105  set([Lausanne-Morges, Campus])
          48   set([Lausanne-Morges, Campus])
```

we see that the user is over-whelmingly females. However, a look at the lower end of such addresses seems alright,

```
Out[807]:                             address  numFemales  numMales  numUsers  \
          3                 poudrière 24; 1950           3         0         3
          61           avenue beaulieu 20; 1004           2         1         3
          23   avenue louis-ruchonnet 31; 1003           2         1         3
          104                eichenweg 12; 1718           2         1         3
          67         rue saint-rochemin 5; 1004           2         1         3

                                subnetworks
          3     set([Campus, Valais Central])
          61   set([Lausanne-Morges, Campus])
          23   set([Lausanne-Morges, Campus])
          104   set([Agglo Fribourg, Campus])
          67   set([Lausanne-Morges, Campus])
```

These particular addresses appear sensible. There could be more than one person living at these addresses who have signed up with the bike system, albeit in different subnetworks. Or may be it is the same person with 2 different sign-ups in two different sub-networks. This raises the question: **How are users registered by the system? One individual = one signup? Or does a user need a sign-up for each subnetwork that she wants to use?** If it is the latter, then the provided *user_ids* become less useful, because the same individual will appear as different users according to the *user_ids*.

```
Out[808]:                           address  numFemales  numMales  numUsers
          18       via lambertenghi 1; 6900          52         1        53
          2288  chemin des falaises 3; 1005          52         0        52
          1349   chemin des berges 12; 1022          41         0        41
          2150     avenue des bains 9; 1007          37         0        37
          332      via monte carmen 4; 6900          33         0        33
          287     route cantonale 33; 1025           25         0        25
          1444      via madonnetta 23; 6900          24         0        24
          1649     place du tunnel 17; 1005          23         0        23
          1826    avenue des bains 11; 1007          23         0        23
          1997        rue de genève 76; 1004         22         0        22
          1801     route cantonale 35; 1025          22         0        22
          2534            via zurigo 1; 6900         20         1        21
```

```
Out[809]:                         address  numFemales  numMales  numUsers
          0      bonne-espérance 28; 1006           1         0         1
          1     37 route cantonnale; 1025           1         0         1
          2     avenue de la dôle 4; 1005           1         0         1
          3             abbesses 21; 2012           1         0         1
          4  chemin de ponfilet 100; 1093           0         1         1
```