

Welcome to the Workshop on Web and Social Media Data for Demographic Research

Emilio Zagheni
with Charles Lanfear, Tom Le Grand,
Julio Ortega et al.

ALAP and ABEP Conference
Foz do Iguacu, PR, Brazil, October 17, 2016

Organization: IUSSP Panel on Big Data and Population
Processes, with support from PAA and Hewlett Foundation

IUSSP Panel on Big Data and Population Processes



Bogdan State



Emilio Zagheni



Francesco Billari



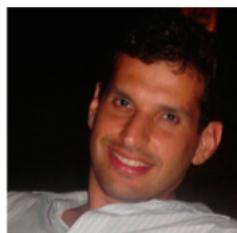
Emmanuel Letouze



Guy Abel



Melinda Mills



Cassio Turra



Mary Ellen Zuppan (IUSSP)



Tom Le Grand
(Council Liaison)



Paul Monet
(IUSSP Secretariat)

Recent and planned Panel activities

► **Hands-on Training Workshops**

- ▶ at Population Association of America, (Washington, D.C., April 2016)
- ▶ at European Population Conference (Mainz, August 2016)
- ▶ at Brazilian+Latin American Population Conference (Foz do Iguacu, October 2016)

► **Mini-conferences/Research Workshops**

- ▶ at the International Conference on Web and Social Media (May 2016, Cologne)
- ▶ at Social Informatics Conference (Bellevue, Seattle, November 2016)

Recent and planned Panel activities

► **Hands-on Training Workshops**

- ▶ at Population Association of America, (Washington, D.C., April 2016)
- ▶ at European Population Conference (Mainz, August 2016)
- ▶ at Brazilian+Latin American Population Conference (Foz do Iguacu, October 2016)

► **Mini-conferences/Research Workshops**

- ▶ at the International Conference on Web and Social Media (May 2016, Cologne)
- ▶ at Social Informatics Conference (Bellevue, Seattle, November 2016)
- ▶ More info at: <http://iusspp.org/en/panel/big-data-and-population-processes>

Who is in today's group? Who is sitting by me?



General goals for today

We will address three broad questions:

1. How to collect and analyze social media data?
 2. How can demographers benefit from and contribute to social media research?
 3. What are some examples of research questions in the emerging area of ‘digital demography’?
- ⇒ Emphasis on having fun with hands-on activities

Plan for today

- ▶ Introduction to APIs (Application Programming interfaces) and hands-on example with Face++
- ▶ Collecting Twitter data using R
- ▶ Collecting data from Facebook public pages + simple sentiment analysis
- ▶ Making sense of the data using Demographic methods

Not everything runs smoothly the first time we use it...



"Hit Any Key To Continue"

Materials for the workshop

- ▶ Materials for this workshop are available on Github at
<http://bit.ly/IUSSP-IGU>
- ▶ Please download the material as a zipped folder and unzip the folder

Today we will look at data like this...

Home Moments Notifications Messages Twitter

Search Twitter



TWEETS 239 FOLLOWING 72 FOLLOWERS 6.87M LIKES 3

President Obama  @POTUS
Dad, husband, and 44th President of the United States. Tweets may be archived: wh.gov/privacy.

Washington, D.C.

WhiteHouse.gov

Tweets Tweets & replies Photos & videos

 President Obama @POTUS · Mar 20
¿Que bolá Cuba? Just touched down here, looking forward to meeting and hearing directly from the Cuban people.

34K 58K ***

Who to follow · Refresher

 PyData @PyData
[Follow](#)

 Chris Uggen @
Followed by Ch

[Follow](#)

or tweets like this one:

 **Barack Obama** 
@BarackObama

 Follow

Four more years.
pic.twitter.com/bAJE6Vom

◀ Reply  Retweet  Favorite  More



783,489 RETWEETS 295,026 FAVORITES



11:16 PM - 6 Nov 12

Flag media

credit: Pablo Barbera

Tweets are stored in JSON format:

```
{ "created_at": "Wed Nov 07 04:16:18 +0000 2012",
  "id": 266031293945503744,
  "text": "Four more years. http://t.co/bAJE6Vom",
  "source": "web",
  "user": {
    "id": 813286,
    "name": "Barack Obama",
    "screen_name": "BarackObama",
    "location": "Washington, DC",
    "description": "This account is run by Organizing for Action staff.  
Tweets from the President are signed -bo.",
    "url": "http://t.co/8aj56Jcemr",
    "protected": false,
    "followers_count": 40873124,
    "friends_count": 654580,
    "listed_count": 202495,
    "created_at": "Mon Mar 05 22:08:25 +0000 2007",
    "time_zone": "Eastern Time (US & Canada)",
    "statuses_count": 10687,
    "lang": "en" },
  "coordinates": null,
  "retweet_count": 783488,
  "favorite_count": 295026,
  "lang": "en"
}
```

credit: Pablo Barbera

Tweets are stored in JSON format:

```
{ "created_at": "Wed Nov 07 04:16:18 +0000 2012",
  "id": 266031293945503744,
  "text": "Four more years. http://t.co/bAJE6Vom",
  "source": "web",
  "user": {
    "id": 813286,
    "name": "Barack Obama",
    "screen_name": "BarackObama",
    "location": "Washington, DC",
    "description": "This account is run by Organizing for Action staff.  
Tweets from the President are signed -bo.",
    "url": "http://t.co/8aJ56Jcemr",
    "protected": false,
    "followers_count": 40873124,
    "friends_count": 654580,
    "listed_count": 202495,
    "created_at": "Mon Mar 05 22:08:25 +0000 2007",
    "time_zone": "Eastern Time (US & Canada)",
    "statuses_count": 10687,
    "lang": "en" },
  "coordinates": null,
  "retweet_count": 783488,
  "favorite_count": 295026,
  "lang": "en"
}
```

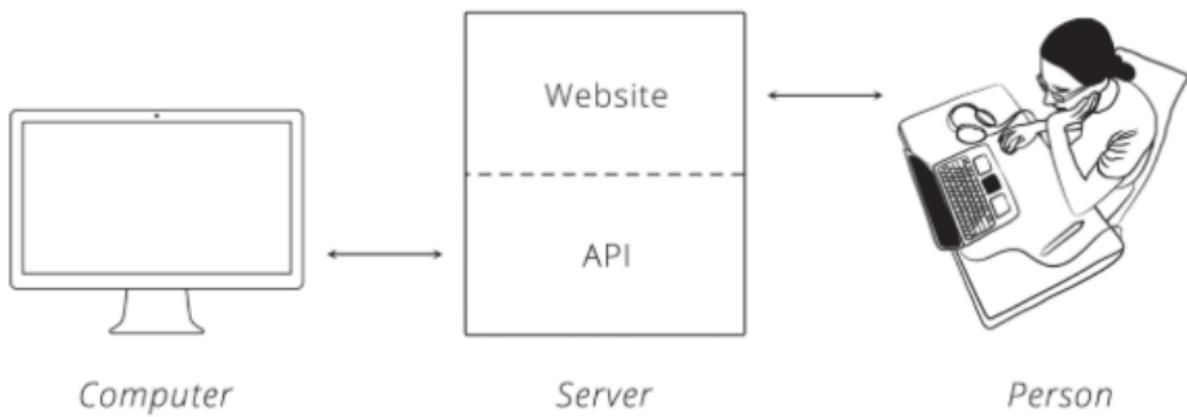
Tweets are stored in JSON format:

```
{ "created_at": "Wed Nov 07 04:16:18 +0000 2012",
  "id": 266031293945503744,
  "text": "Four more years. http://t.co/bAJE6Vom",
  "source": "web",
  "user": {
    "id": 813286,
    "name": "Barack Obama",
    "screen_name": "BarackObama",
    "location": "Washington, DC",
    "description": "This account is run by Organizing for Action staff.  
Tweets from the President are signed -bo.",
    "url": "http://t.co/8aJ56Jcemr",
    "protected": false,
    "followers_count": 40873124,
    "friends_count": 654580,
    "listed_count": 202495,
    "created_at": "Mon Mar 05 22:08:25 +0000 2007",
    "time_zone": "Eastern Time (US & Canada)",
    "statuses_count": 10687,
    "lang": "en" },
  "coordinates": null,
  "retweet_count": 783488,
  "favorite_count": 295026,
  "lang": "en"
}
```

Values can
be lists

How do we access these data?

How do we access these data?



Credit: Brian Cooksey

Example of integrated Websites

Example of integrated Websites

yelp Find tacos, cheap dinner, Max's Near Seattle, WA

Home About Me Write a Review Find Friends Messages Talk Events

University of Washington

★ ★ ★ ★ 114 reviews Details

Colleges & Universities Edit

Map data ©2015 Google

1400 NE Campus Pkwy
Seattle, WA 98195
University District
Get Directions
Call (206) 543-2100
Email washington.edu

Write a Review Add Photo Share Bookmark

See all 260 photos

"I've seen a handful of campuses and I have to say, I was incredibly impressed when I first visited UW." in 52 reviews

"Sitting under the cherry blossoms each spring on the Quad....like a snowstorm only with cherry blossom petals." in 16 reviews

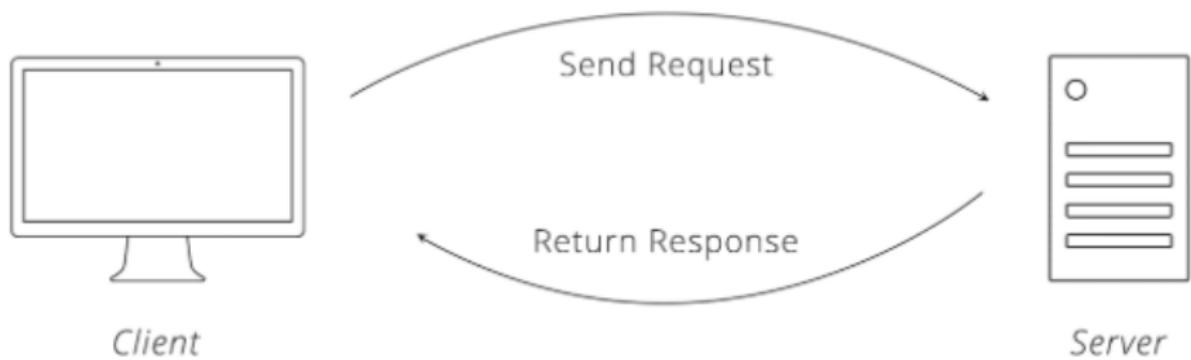
"Fortunately, I came during early spring when the cherry blossoms were still there." in 9 reviews

Edit business info Work here? Claim this business

Help fight food deserts. DRINK DO

Communication typically happens via HTTP

Communication typically happens via HTTP



Credit: Brian Cooksey

Authentication

Authentication

- ▶ We have all registered for a website before (e.g., Facebook, Google, Dropbox, etc.):

Authentication

- ▶ We have all registered for a website before (e.g., Facebook, Google, Dropbox, etc.):
- ▶ Logging-in with a username and password is an example of authentication

Authentication

- ▶ We have all registered for a website before (e.g., Facebook, Google, Dropbox, etc.):
- ▶ Logging-in with a username and password is an example of authentication
- ▶ A different example: The developer of a game app can use the Dropbox API to let users store their saved games in the Dropbox cloud

Authentication

- ▶ We have all registered for a website before (e.g., Facebook, Google, Dropbox, etc.):
- ▶ Logging-in with a username and password is an example of authentication
- ▶ A different example: The developer of a game app can use the Dropbox API to let users store their saved games in the Dropbox cloud
- ▶ Would you share your Dropbox username and password with the game developer?

API Key Authentication

API Key Authentication

- ▶ It is a technique to overcome the weakness of sharing credentials

API Key Authentication

- ▶ It is a technique to overcome the weakness of sharing credentials
- ▶ The key is usually a long series of letters and numbers that is distinct from the account owner's login password

API Key Authentication

- ▶ It is a technique to overcome the weakness of sharing credentials
- ▶ The key is usually a long series of letters and numbers that is distinct from the account owner's login password
- ▶ The owner gives the key to the application

API Key Authentication

- ▶ It is a technique to overcome the weakness of sharing credentials
- ▶ The key is usually a long series of letters and numbers that is distinct from the account owner's login password
- ▶ The owner gives the key to the application
- ▶ The owner can restrict administrative privileges associated with the key

Let's consider an example: FACE++

Let's consider an example: FACE++

- ▶ Face++ provides functions for face detection

Let's consider an example: FACE++

- ▶ Face++ provides functions for face detection
- ▶ Useful to generate estimates of demographic attributes from face images

Let's consider an example: FACE++

- ▶ Face++ provides functions for face detection
- ▶ Useful to generate estimates of demographic attributes from face images
- ▶ Example: detect age, sex and race of social media users based on their profile pictures

Let's consider an example: FACE++

- ▶ Face++ provides functions for face detection
- ▶ Useful to generate estimates of demographic attributes from face images
- ▶ Example: detect age, sex and race of social media users based on their profile pictures
- ▶ URL: <http://www.faceplusplus.com>

Go to www.faceplusplus.com/demo-detect

www.faceplusplus.com/demo-detect/

...istics blog Computer se...| UW Today Call: 201-877-973-3346 A Beginner's...w Rathbone Cloudera QuickStart VM Required Tra...ed Training O'Reilly

Demo detect | Face++

SIGN IN SIGN UP LANGUAGE ▾

FACE++

Home Tech + Service Examples Demo Dev Center Research

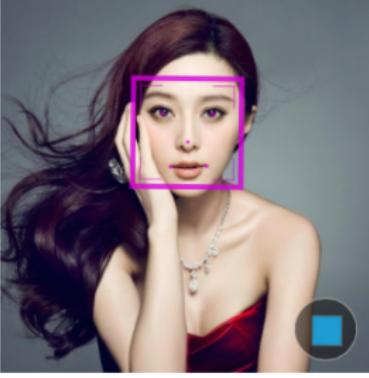


Demo

- Face Detection >
- Face Search >
- Face Landmark >
- Face Mask >
- Interactive Demo >

Tips:

Select sample image, paste picture URL, or upload local pictures for face detection demo. You can also use the Chrome browser for taking photos online.



URL

REST URL:
`http://api.us.faceplusplus.com/v2/detection/detect?api_key=DEMO_KEY&api_secret=DEMO_SECRET&url=http://3AX2F2z2fwww.faceplusplus.com%2fwp-content%2f2t`

RESPONSE JSON:

```
{ "face": [ { "attribute": { "age": { "range": 5, "value": 24 } }, "gender": { "confidence": 99.9999, "value": "Female" }, "glass": { "confidence": 99.4157, "value": "None" }, "pose": { "pitch_angle": { }}
```

Example: what are the demographic attributes of Ronald Lee, the 2016 IUSSP Laureate?



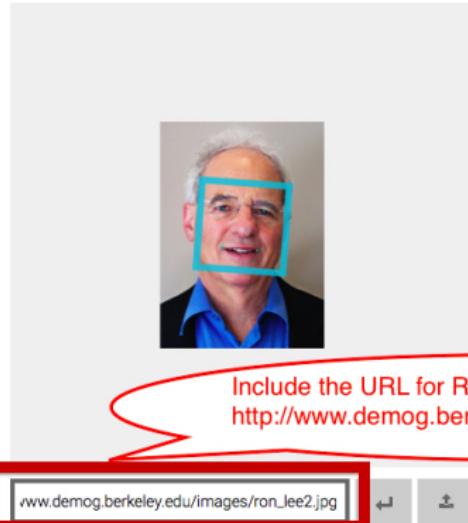
http://www.demog.berkeley.edu/images/ron_lee2.jpg

Demo

- Face Detection >
- Face Search >
- Face Landmark >
- Face Mask >
- Interactive Demo >

Tips:

Select sample image, paste picture URL, or upload local pictures for face detection demo. You can also use the Chrome browser for taking photos online.



REST URL:

```
http://api.us.faceplusplus.com/v2/detection/detect?api_key=DEMO_KEY&api_secret=DEMO_SECRET&url=http%3A%2F%2Fwww.demog.berkeley.edu%2Fimages%2Fron_lee2.jpg&attribute=age%2Cgender%2Crace%2Csmiling%2Cpose%2Cglass
```

RESPONSE JSON:

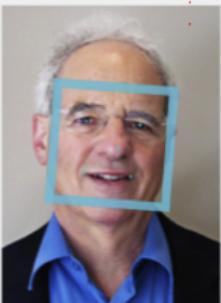
```
{  
  "face": [  
    {  
      "attribute": {  
        "age": {  
          "range": 10,  
          "value": 66  
        },  
        "gender": {  
          "confidence": 99.736,  
          "value": "Male"  
        }  
      }  
    }  
  ]  
}
```

Include the URL for Ron's image here:
http://www.demog.berkeley.edu/images/ron_lee2.jpg

www.demog.berkeley.edu/images/ron_lee2.jpg

```
  "pose": {  
    "roll": -0.00064,  
    "value": -0.00064  
  },  
  "roll_angle": {  
    "value": -0.00064  
  }  
}
```

Response from API with various attributes of the image



www.demog.berkeley.edu/images/ron_lee2.jpg



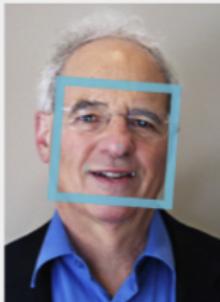
REST URL:

`http://api.us.faceplusplus.com/v2/detection/detect?api_key=DEMO_KEY&api_secret=DEMO_SECRET&url=http%3A%2F%2Fwww.demog.berkeley.edu%2Fimages%2Fron_lee2.jpg&attribute=age%2Cgender%2Crace%2Csmiling%2Cpose%2Cglass`

RESPONSE JSON:

```
{  
  "face": [  
    {  
      "attribute": {  
        "age": {  
          "range": 10,  
          "value": 66  
        },  
        "gender": {  
          "confidence": 99.736,  
          "value": "Male"  
        },  
        "glass": {  
          "confidence": 81.4425,  
          "value": "None"  
        },  
        "pose": {  
          "pitch_angle": {  
            "value": -0.000064  
          },  
          "roll_angle": {  
            "value": 0.000001  
          }  
        }  
      }  
    }  
  ]  
}
```

We would get the same response using this URL



www.demog.berkeley.edu/images/ron_lee2.jpg



REST URL:

```
http://api.us.faceplusplus.com/v2/detection/detect?api_key=DEMO_KEY&api_secret=DEMO_SECRET&url=http%3A%2F%2Fwww.demog.berkeley.edu%2Fimages%2Fron_lee2.jpg&attribute=age%2Cgender%2Crace%2Csmiling%2Cpose%2Cglass
```

RESPONSE JSON:

```
{
  "face": [
    {
      "attribute": {
        "age": {
          "range": 10,
          "value": 66
        },
        "gender": {
          "confidence": 99.736,
          "value": "Male"
        },
        "glass": {
          "confidence": 81.4425,
          "value": "None"
        },
        "pose": {
          "pitch_angle": {
            "value": -0.000064
          },
          "roll_angle": {
            "value": 0.000001
          }
        }
      }
    }
  ]
}
```



REST URL:

```
http://api.us.faceplusplus.com/v2/detection/detect?api_key  
=DEMO_KEY&api_secret=DEMO_SECRET&url=http%3A%2F%2Fwww.dem  
og.berkeley.edu%2Fimages%2Fron_lee2.jpg&attribute=age%2Cg  
ender%2Crace%2Csmiling%2Cpose%2Cglass
```

RESPONSE JSON:

```
{  
  "face": [  
    {  
      "attribute": {  
        "age": {  
          "range": 10,
```



Your Authentication Key and Secret

REST URL:

```
http://api.us.faceplusplus.com/v2/detection/detect?api_key  
[REDACTED] DEMO_KEY [REDACTED] api_secret [REDACTED] DEMO SECRET url=http%3A%2Fwww.dem  
og.berkeley.edu%2Fimages%2Fron_lee2.jpg&attribute=age%2Cg  
ender%2Crace%2Csmiling%2Cpose%2Cglass
```

RESPONSE JSON:

```
[{"face": [  
  {  
    "attribute": {  
      "age": {  
        "range": 10,
```



REST URL:

`http://api.us.faceplusplus.com/v2/detection/detect?api_key
=DEMO_KEY&api_secret=DEMO_SECRET&url=http%3A%2F%2Fwww.dem
og.berkeley.edu%2Fimages%2Ffrontee2.jpg&attribute=age%2Cg
ender%2Crace%2Csmiling%2Cpose%2Cglass`

Image URL

RESPONSE JSON:

```
{  
  "face": [  
    {  
      "attribute": {  
        "age": {  
          "range": 10,
```



REST URL:

http://api.us.faceplusplus.com/v2/detection/detect?api_key
=DEMO_KEY&api_secret=DEMO_SECRET&url=http%3A%2F%2Fwww.dem
og.berkeley.edu%2Fimages%2Fron_lee2.jpg&attribute[age%2Cg
ender%2Crace%2Csmiling%2Cpose%2Cglass

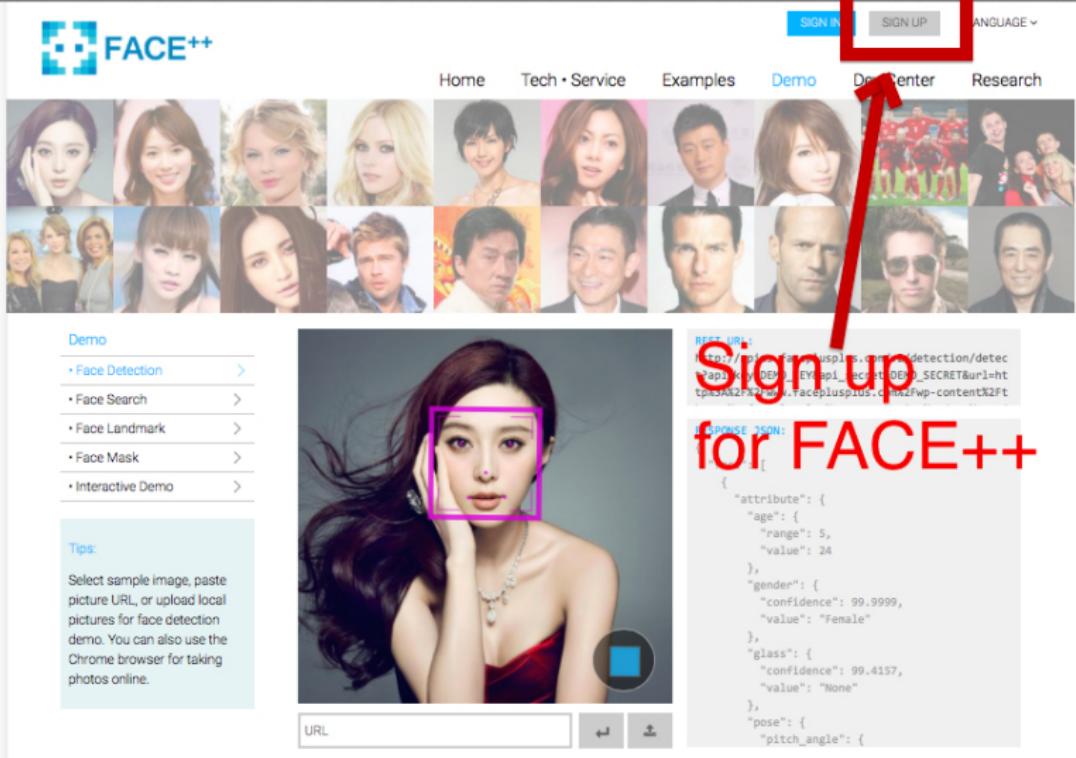
RESPONSE JSON:

```
{  
  "face": [  
    {  
      "attribute": {  
        "age": {  
          "range": 10,
```

Attributes we
request

Two key steps to use the API efficiently

1. We need to register an APP with Face++
2. We need to automate the process of extracting data



**Sign up
for FACE++**

[SIGN IN](#)[SIGN UP](#)[Dev Center](#)[My App](#)[Getting Started](#)[API Docs](#)[Dev](#)

Sign up

Email



1. Insert your info
2. Validate your email

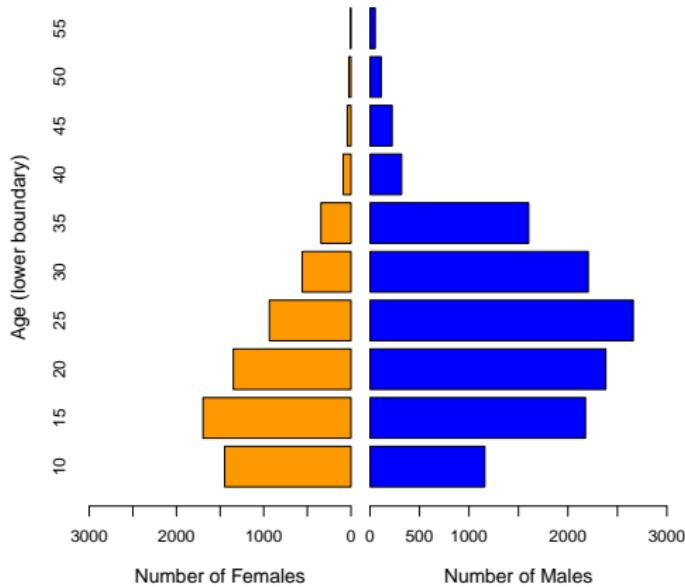
Username

Password

[SIGN UP](#)

App Name	my_PAA2016_app
API Key	df7de419c67beba9123b17480527adfd
API Secret	EgiZTXLm5e85fR02JdDITEn622rG81Xu RESET API_SECRET ?
API Url	api.us.faceplusplus.com ?
Start Date	March 23, 2016, 2:34:18 pm
App Information	test the software for demographic research
App Status	Developer [Upgrade]
API Server	Amazon(US)
App Type	Education
Platform	Web
Local Detector	IOS ANDROID

Motivation: Face++ applied to profile pictures of Twitter users



Source: Zagheni, Garimella et al. (2014). Inferring International and Internal Migration Patterns from Twitter Data.

Let's look at some R code