STRAVA SNA Project

Jacob Verdesi, David Bignelli and Nicolas Chaudron

Business Understanding

Your just moved to Rochester for job at a upcoming bike manufacturer as a social network analyst.

They want you to find a community suitable to market there new mountain bike to, with no budget.

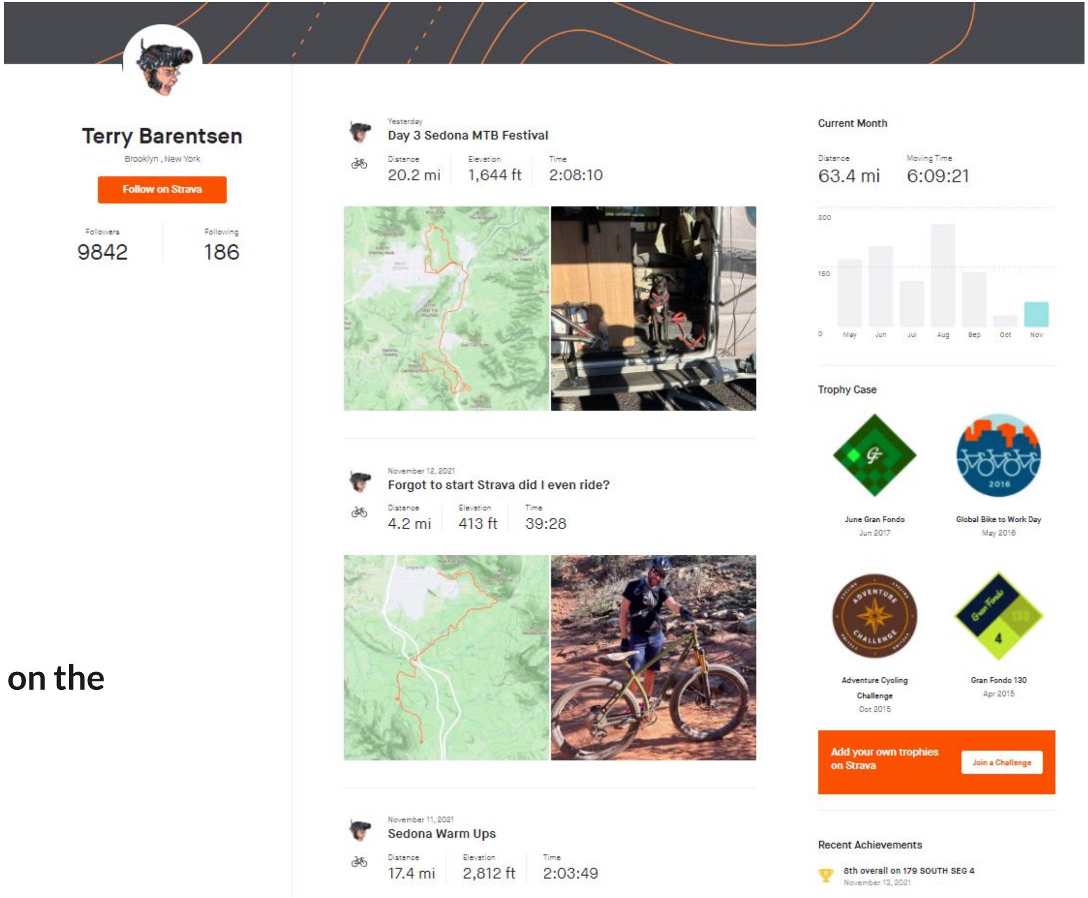
Your idea is to identify local communities of Strava users using the publicly available Strava data.

STRAVA™

What is Strava

A Community for Athletes

- Post activities to public and friends
- 30 M+ users
- Cycling, Running, Walking, ect..
- Performance Metrics
- Clubs and Trophies
- Segments to Compete against others on the leaderboard!
- Free version or Subscription



Terry Barentsen
Brooklyn , New York

Followers 9842
Following 186

Yesterday
Day 3 Sedona MTB Festival
Distance 20.2 mi | Elevation 1,644 ft | Time 2:08:10

November 12, 2021
Forgot to start Strava did I even ride?
Distance 4.2 mi | Elevation 413 ft | Time 39:28

November 11, 2021
Sedona Warm Ups
Distance 17.4 mi | Elevation 2,812 ft | Time 2:03:49

Current Month
Distance 63.4 mi
Moving Time 6:09:21

Trophy Case
June Gran Fondo
Jun 2017
Global Bike to Work Day
May 2016
Adventure Cycling Challenge
Oct 2015
Gran Fondo 130
Apr 2015

Add your own trophies on Strava

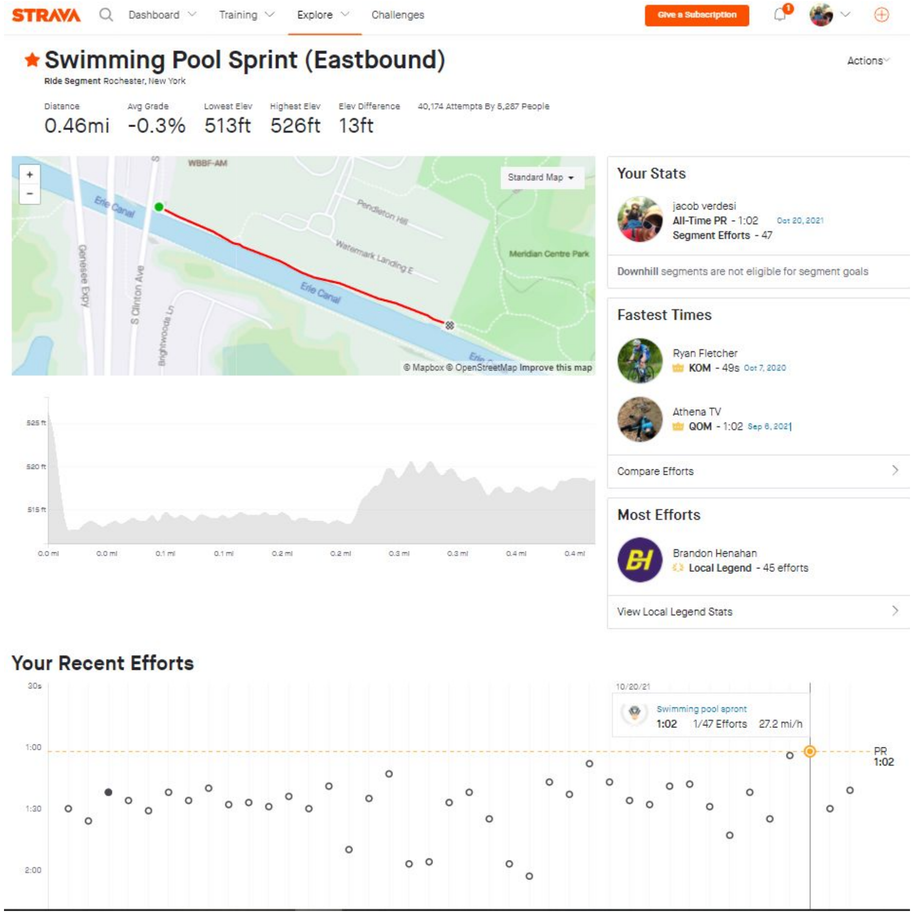
Recent Achievements
8th overall on 179 SOUTH SEG 4
November 13, 2021

What is a Segment

They are portions of road or trail created by members where athletes can compare times.

To be King of the Mountain (KOM), Queen of the Mountain (QOM), you have to have the fastest time on a segment.

Plenty of statistics about the segment as well as leaderboard to compete for top times.



Swimming Pool Sprint (Eastbound)

Ride Segment Rochester, New York

Distance 0.46mi · Avg Grade -0.3% · Lowest Elev 513ft · Highest Elev 526ft · Elev Difference 13ft · 40,174 Attempts By 5,287 People

Your Stats

jacob verdesi
All-Time PR - 1:02 Oct 20, 2021
Segment Efforts - 47

Downhill segments are not eligible for segment goals

Fastest Times

Ryan Fletcher
KOM - 49s Oct 7, 2020

Athena TV
QOM - 1:02 Sep 6, 2021

Compare Efforts

Most Efforts

Brandon Henahan
Local Legend - 45 efforts

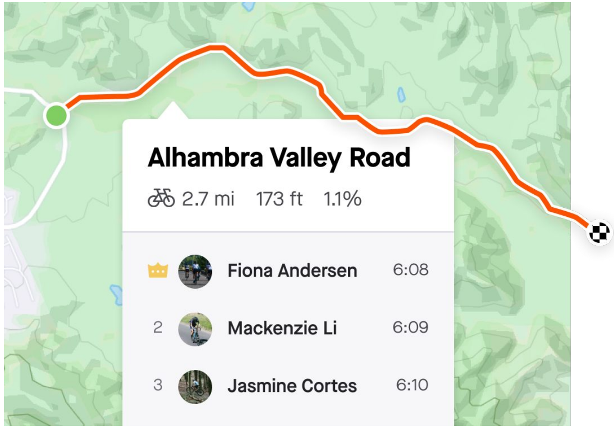
View Local Legend Stats

Your Recent Efforts

Swimming pool sprint
1:02 1/47 Efforts 27.2 mi/h

Data Understanding

To try and identify communities, we want to graph the relationships of segments through the number of common people that completed those segments.



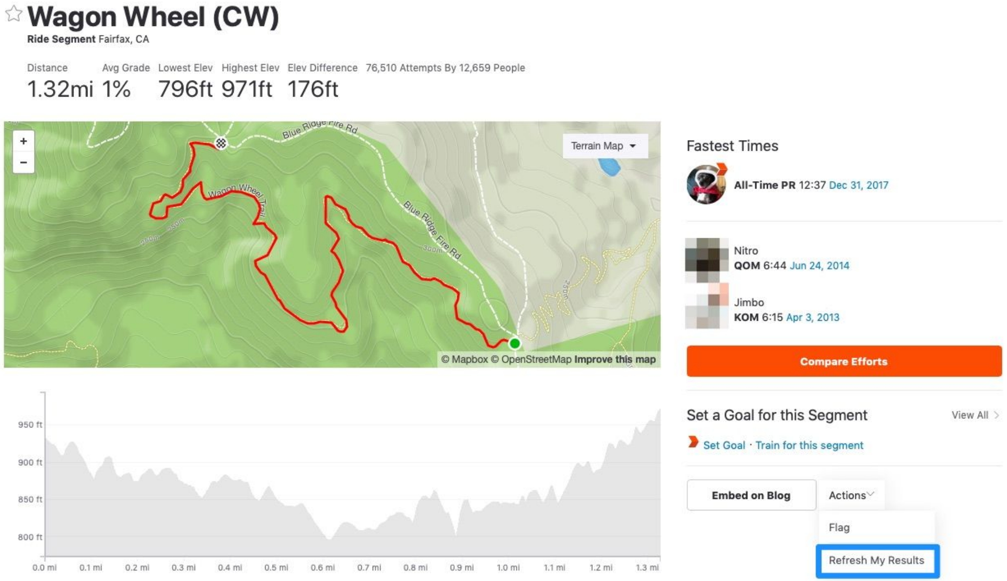
Alhambra Valley Road

2.7 mi 173 ft 1.1%

👑	Fiona Andersen	6:08
2	Mackenzie Li	6:09
3	Jasmine Cortes	6:10

Data Collection

- Limitations
  - Rate Limiting
  - Data Availability: API vs App
- Two Methods
  - API
  - Web Scraper



Wagon Wheel (CW)

Ride Segment Fairfax, CA

Distance	Avg Grade	Lowest Elev	Highest Elev	Elev Difference	76,510 Attempts By 12,659 People
1.32mi	1%	796ft	971ft	176ft	

Fastest Times

All-Time PR 12:37 Dec 31, 2017

Nitro
QOM 6:44 Jun 24, 2014

Jimbo
KOM 6:15 Apr 3, 2013

Set a Goal for this Segment

API

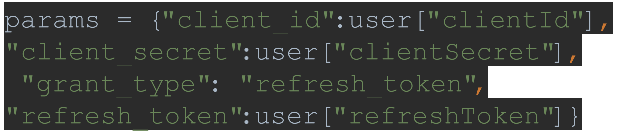
Challenges

- Developer Account
- OAuth 2.0
- Rate Limiter

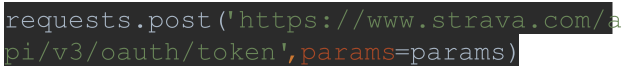
Tools

- PostMan
- Python Requests Package
- Pandas

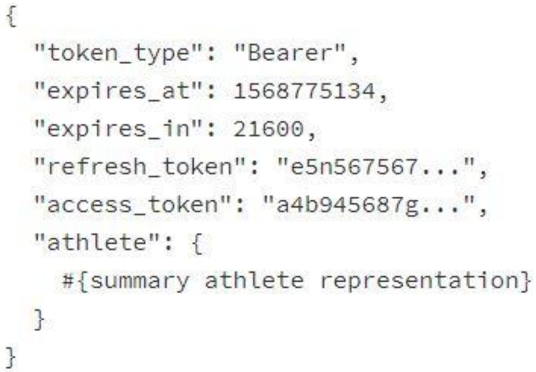
Auth Request



```
params = {"client_id":user["clientId"],
"client_secret":user["clientSecret"],
 "grant_type": "refresh_token",
"refresh_token":user["refreshToken"]}
```

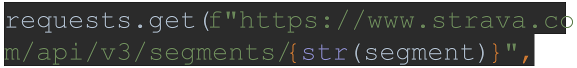


```
requests.post('https://www.strava.com/api/v3/oauth/token',params=params)
```

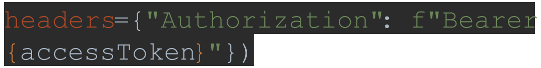


```
{
  "token_type": "Bearer",
  "expires_at": 1568775134,
  "expires_in": 21600,
  "refresh_token": "e5n567567...",
  "access_token": "a4b945687g...",
  "athlete": {
    #{summary athlete representation}
  }
}
```

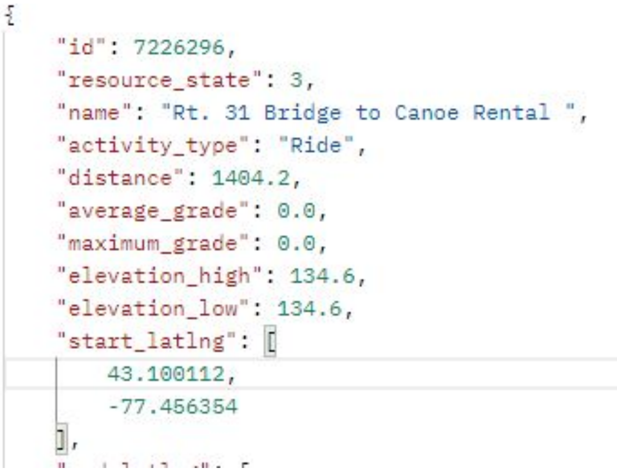
Segment Metadata Request



```
requests.get(f"https://www.strava.com/api/v3/segments/{str(segment)}",
```

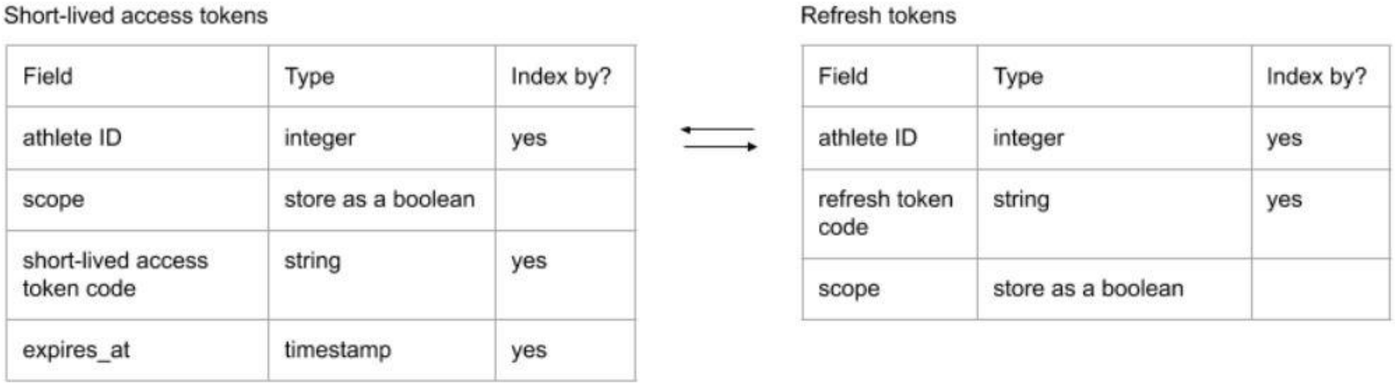


```
headers={"Authorization": f"Bearer {accessToken}"})
```



```
{
  "id": 7226296,
  "resource_state": 3,
  "name": "Rt. 31 Bridge to Canoe Rental ",
  "activity_type": "Ride",
  "distance": 1404.2,
  "average_grade": 0.0,
  "maximum_grade": 0.0,
  "elevation_high": 134.6,
  "elevation_low": 134.6,
  "start_latlng": [
    43.100112,
    -77.456354
  ],
```

API Authentication



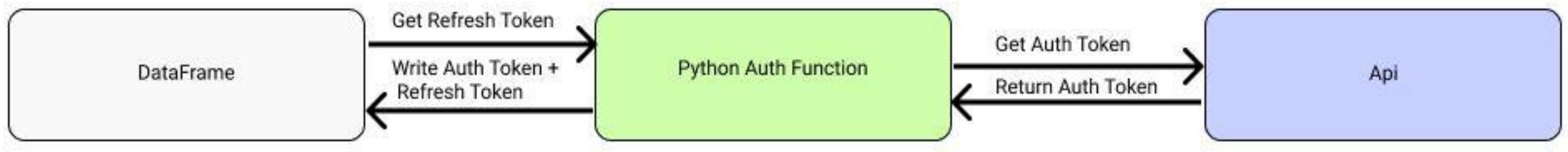
Short-lived access tokens

Field	Type	Index by?
athlete ID	integer	yes
scope	store as a boolean	
short-lived access token code	string	yes
expires_at	timestamp	yes

Refresh tokens

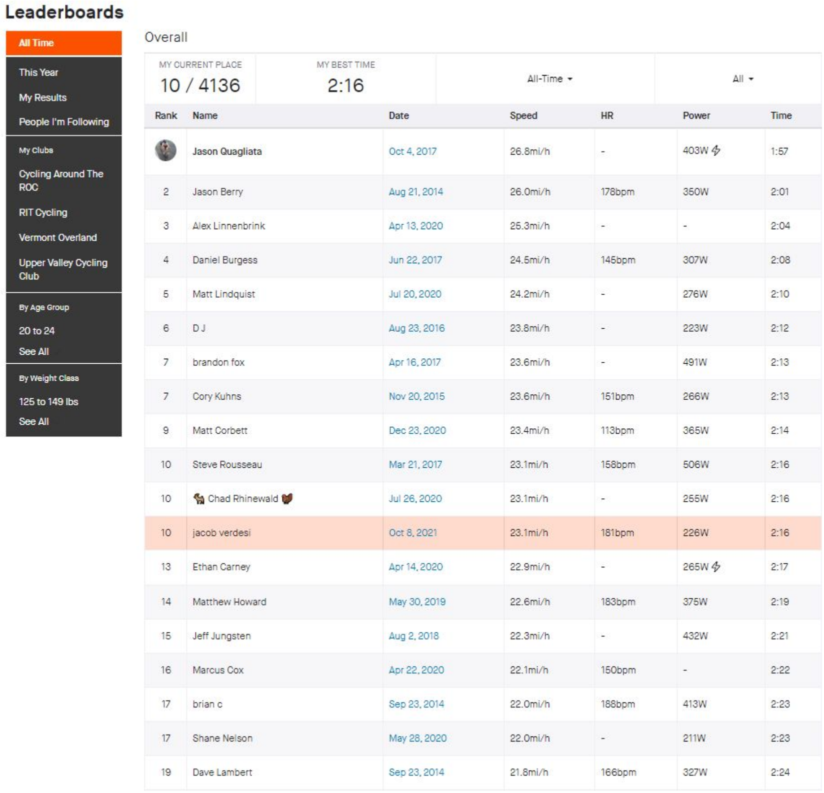
Field	Type	Index by?
athlete ID	integer	yes
refresh token code	string	yes
scope	store as a boolean	

Auth Implementation



Web Scraper

- The public leaderboards' data was not available via the API.
- This Data allows us to connect the segments and the athletes
- Iterated over segments finding each Athlete's unique activity_id by segment
- Parse Html to search for <table> then convert to pandas Dataframe



Overall

MY CURRENT PLACE 10 / 4136

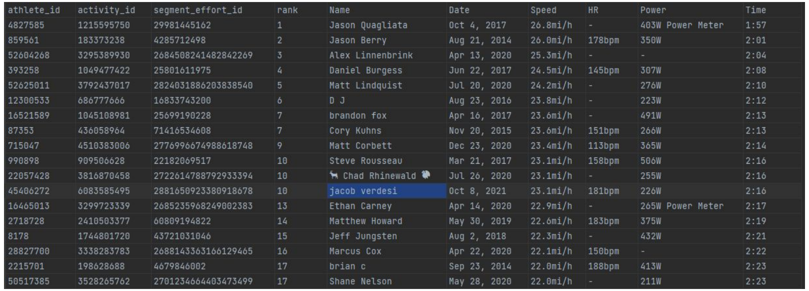
MY BEST TIME 2:16

Rank	Name	Date	Speed	HR	Power	Time
	Jason Quagliata	Oct 4, 2017	26.8mi/h	-	403W ⚡	1:57
2	Jason Berry	Aug 21, 2014	26.0mi/h	178bpm	350W	2:01
3	Alex Linnenbrink	Apr 13, 2020	25.3mi/h	-	-	2:04
4	Daniel Burgess	Jun 22, 2017	24.5mi/h	145bpm	307W	2:08
5	Matt Lindquist	Jul 20, 2020	24.2mi/h	-	276W	2:10
6	D J	Aug 23, 2016	23.8mi/h	-	223W	2:12
7	brandon fox	Apr 16, 2017	23.6mi/h	-	491W	2:13
7	Cory Kuhns	Nov 20, 2015	23.6mi/h	151bpm	266W	2:13
9	Matt Corbett	Dec 23, 2020	23.4mi/h	113bpm	365W	2:14
10	Steve Rousseau	Mar 21, 2017	23.1mi/h	158bpm	506W	2:16
10	🐕 Chad Rhinewald 🦃	Jul 26, 2020	23.1mi/h	-	255W	2:16
10	jacob verdesi	Oct 8, 2021	23.1mi/h	181bpm	226W	2:16
13	Ethan Carney	Apr 14, 2020	22.9mi/h	-	265W ⚡	2:17
14	Matthew Howard	May 30, 2019	22.6mi/h	183bpm	375W	2:19
15	Jeff Jungsten	Aug 2, 2018	22.3mi/h	-	432W	2:21
16	Marcus Cox	Apr 22, 2020	22.1mi/h	150bpm	-	2:22
17	brian c	Sep 23, 2014	22.0mi/h	188bpm	413W	2:23
17	Shane Nelson	May 28, 2020	22.0mi/h	-	211W	2:23
19	Dave Lambert	Sep 23, 2014	21.8mi/h	166bpm	327W	2:24

Data Preparation

Web Scraper Segments

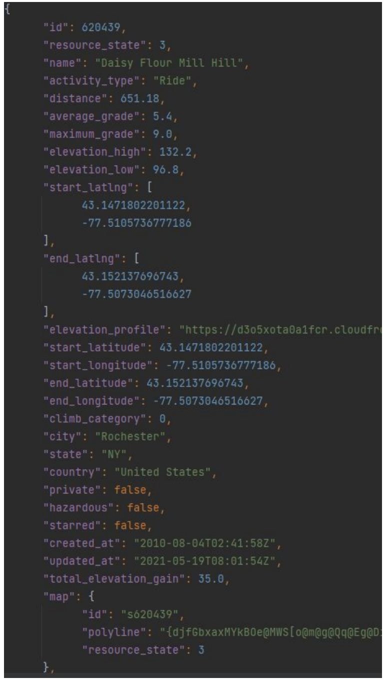
- Csv foreach segment_id (1000 files)
- Series of Athlete_id between (1-6000 rows)
- Drop Performance indicators , Name, Date
- Remove Hazardous segments



athlete_id	activity_id	segment_effort_id	rank	Name	Date	Speed	HR	Power	Time
4827585	1215595750	29981445162	1	Jason Quagliata	Oct 4, 2017	26.8mi/h	-	403W Power Meter	1:57
859561	183373238	4285712498	2	Jason Berry	Aug 21, 2014	26.0mi/h	178bpm	350W	2:01
52604268	3295389930	2684508241482842269	3	Alex Linnenbrink	Apr 13, 2020	25.3mi/h	-	-	2:04
393258	1049477422	25801611975	4	Daniel Burgess	Jun 22, 2017	24.5mi/h	145bpm	307W	2:08
52625011	3792437017	2824031886203838540	5	Matt Lindquist	Jul 20, 2020	24.2mi/h	-	276W	2:10
12300533	686777666	16833743200	6	D J	Aug 23, 2016	23.8mi/h	-	223W	2:12
16521589	1045108981	25699190228	7	brandon fox	Apr 16, 2017	23.6mi/h	-	491W	2:13
87353	436058964	71416534608	7	Cory Kuhns	Nov 20, 2015	23.6mi/h	151bpm	266W	2:13
715047	4510383006	2776996674988618748	9	Matt Corbett	Dec 23, 2020	23.4mi/h	113bpm	365W	2:14
990898	909506628	22182069517	10	Steve Rousseau	Mar 21, 2017	23.1mi/h	158bpm	506W	2:16
22057428	3816870458	2722614788792933394	10	🐐 Chad Rhinewald 🦃	Jul 26, 2020	23.1mi/h	-	255W	2:16
45406272	6083585495	2881650923380918678	10	jacob verdesi	Oct 8, 2021	23.1mi/h	181bpm	226W	2:16
16465013	3299723339	2685235968249002383	13	Ethan Carney	Apr 14, 2020	22.9mi/h	-	265W Power Meter	2:17
2718728	2410503377	60809194822	14	Matthew Howard	May 30, 2019	22.6mi/h	183bpm	375W	2:19
8178	1744801720	43721031046	15	Jeff Jungsten	Aug 2, 2018	22.3mi/h	-	432W	2:21
28827700	3338283783	2688143363166129465	16	Marcus Cox	Apr 22, 2020	22.1mi/h	150bpm	-	2:22
2215701	198628688	4679846002	17	brian c	Sep 23, 2014	22.0mi/h	188bpm	413W	2:23
50517385	3528265762	2701234664403473499	17	Shane Nelson	May 28, 2020	22.0mi/h	-	211W	2:23

Api Segment Metadata

- Json
- Segment_id
- Start_latlng
- Polyline
- Effort_count
- Athlete_count



```
{
      "id": 620439,
      "resource_state": 3,
      "name": "Daisy Flour Mill Hill",
      "activity_type": "Ride",
      "distance": 651.18,
      "average_grade": 5.4,
      "maximum_grade": 9.0,
      "elevation_high": 132.2,
      "elevation_low": 96.8,
      "start_latlng": [
            43.1471802201122,
            -77.5105736777186
      ],
      "end_latlng": [
            43.152137696743,
            -77.5073046516627
      ],
      "elevation_profile": "https://d3o5xota0a1fcr.cloudfr
      "start_latitude": 43.1471802201122,
      "start_longitude": -77.5105736777186,
      "end_latitude": 43.152137696743,
      "end_longitude": -77.5073046516627,
      "climb_category": 0,
      "city": "Rochester",
      "state": "NY",
      "country": "United States",
      "private": false,
      "hazardous": false,
      "starred": false,
      "created_at": "2010-08-04T02:41:58Z",
      "updated_at": "2021-05-19T08:01:54Z",
      "total_elevation_gain": 35.0,
      "map": {
            "id": "s620439",
            "polyline": "{djfGbxaxMYkBOe@MWS[o@m@g@Qq@Eg@D
            "resource_state": 3
      },
```

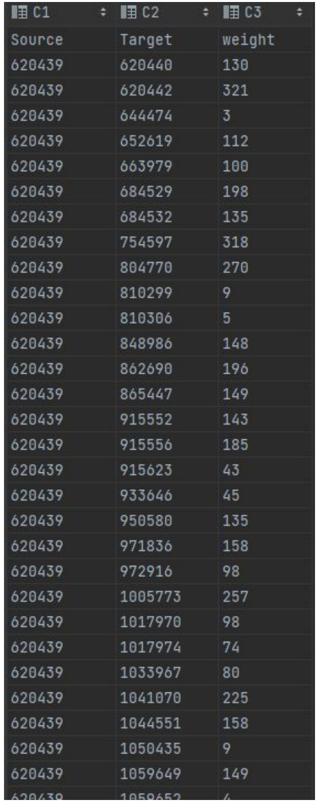
Data Combining

Creating the network

- Create combination of all segments 1026 choose 2 = 525,825 edges
- For each segment combination calculate number of riders in common and use as weight.
- Do not add relationship if no riders in common

Initially tried to create network of athletes with number of segments in common.

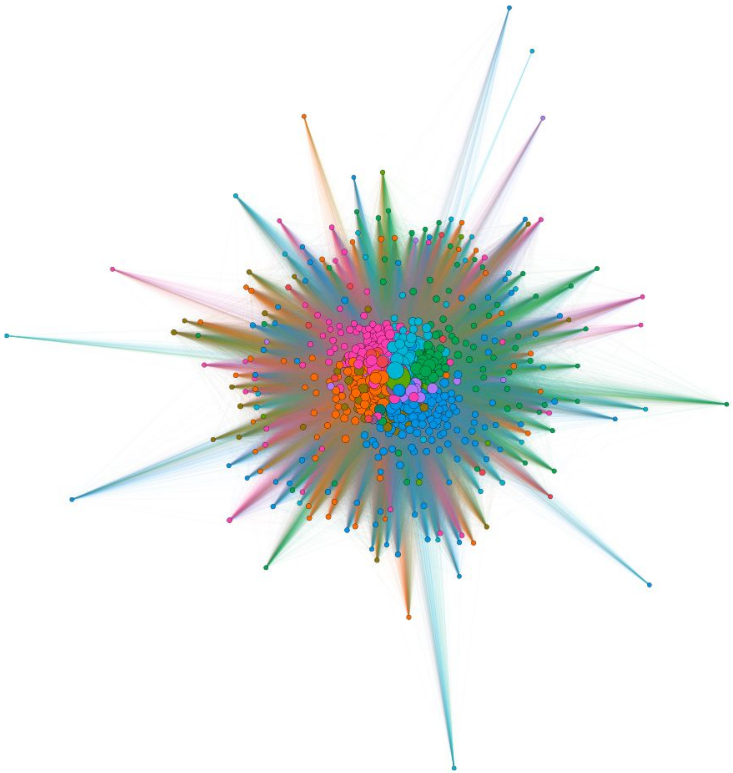
- Network was very large 10,000 unique athletes = 50 million edges
- Tried sampling this using egocentric network / snowball sampling / random
- Became unwieldy in Gephi and validation on map would be much harder and ambiguous



C1	C2	C3
Source	Target	weight
620439	620440	130
620439	620442	321
620439	644474	3
620439	652619	112
620439	663979	100
620439	684529	198
620439	684532	135
620439	754597	318
620439	804770	270
620439	810299	9
620439	810306	5
620439	848986	148
620439	862690	196
620439	865447	149
620439	915552	143
620439	915556	185
620439	915623	43
620439	933646	45
620439	950580	135
620439	971836	158
620439	972916	98
620439	1005773	257
620439	1017970	98
620439	1017974	74
620439	1033967	80
620439	1041070	225
620439	1044551	158
620439	1050435	9
620439	1059649	149
620439	1059652	4

Gephi

- Import Network into Gephi
- Spent a lot of time playing with network statistics / filtering / and layouts
- Modularity proved viable option for clustering segments
- Uses the tie strength
- Export segment modularity classes back into python

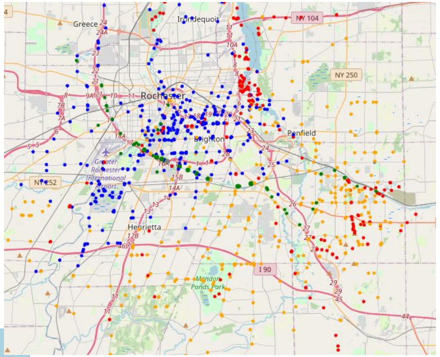


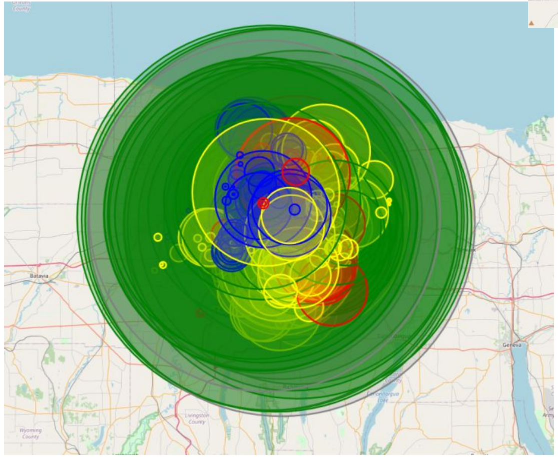
Initial Visualization and Validation

- Used Folium leaflets which is built of OpenStreetMap
- Plot segment lat and lon points
- Color based on modularity
- Size on number of attempts made on segment

Feedback:

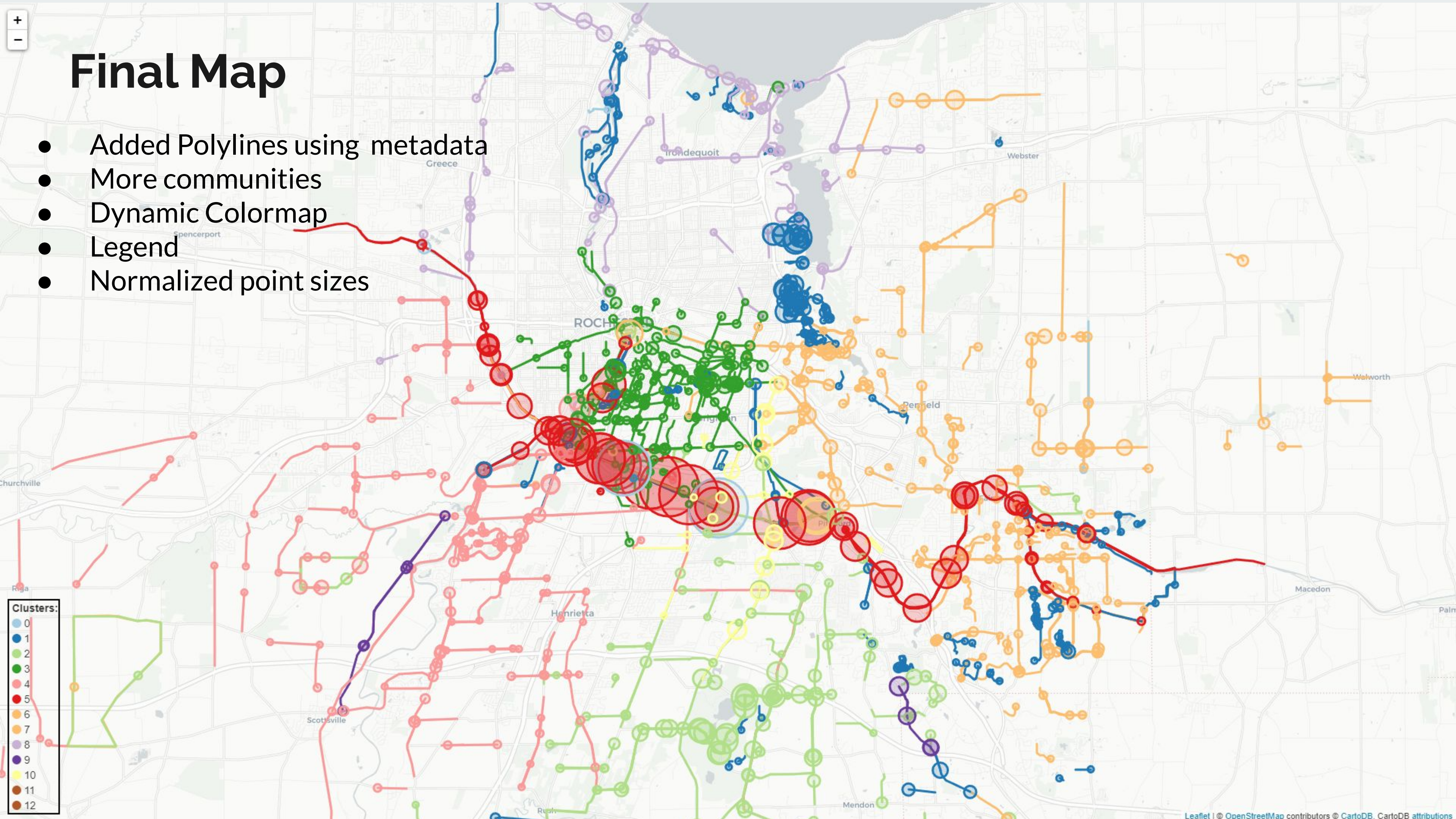
- Some Good Clusters forming
- Large Clusters Dominate
- Did not normalize sizes
- Does Not show full segment
- Hard to read





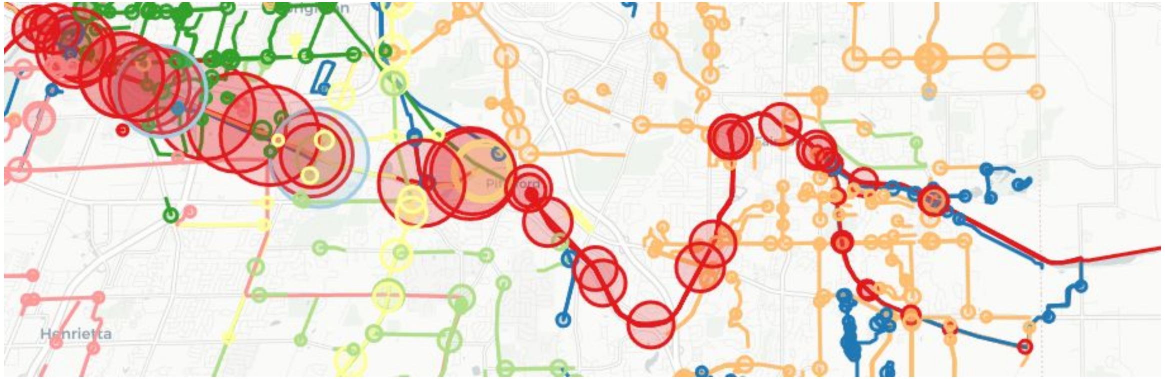
Final Map

- Added Polylines using metadata
- More communities
- Dynamic Colormap
- Legend
- Normalized point sizes

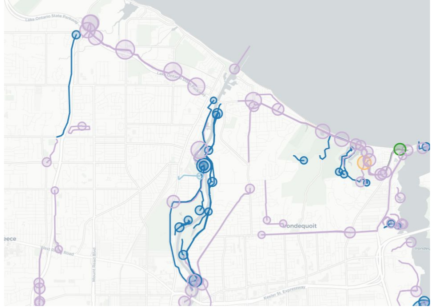


Large Clusters

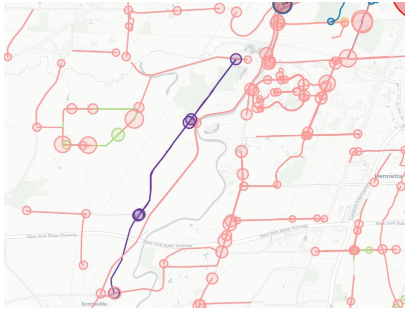
Erie Canal Trail (Red)



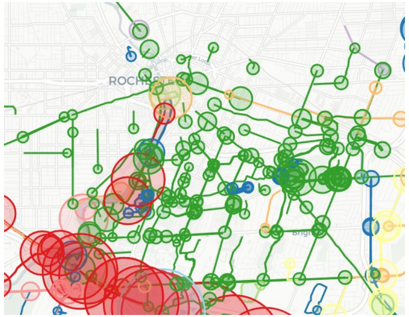
Irondequoit Bay (Light Purple)



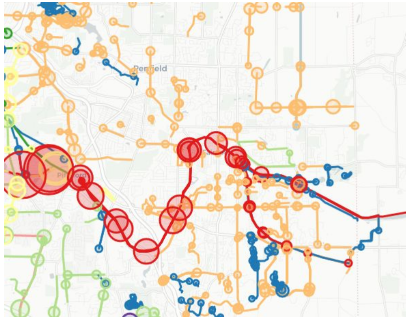
Rit/Henrietta (Pink)



UoR / Central ROC (Green)

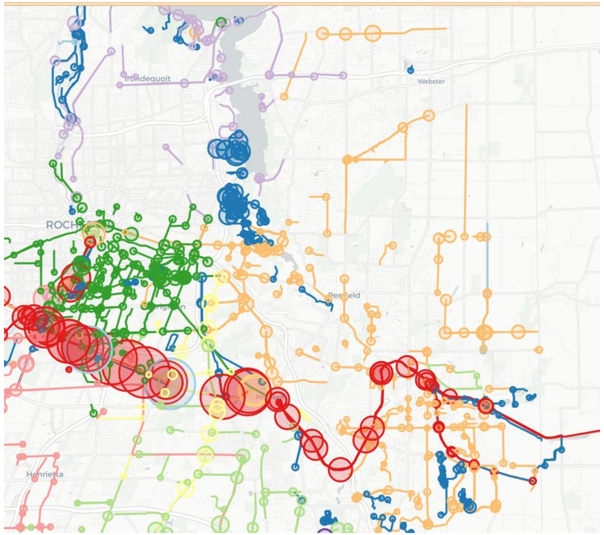


Penfield / Fairport (Gold)

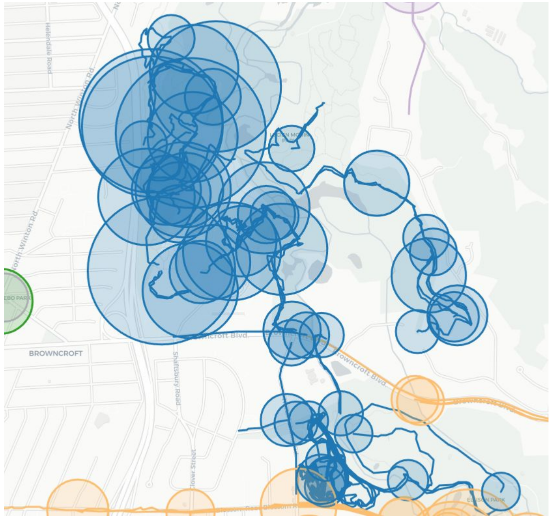


Interesting Clusters

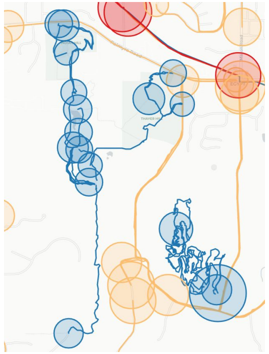
Why is Blue scattered?



Lucian Moran park



Thayer hill



Zoom in to give some hints :

Relatively mountainous and odd polylines

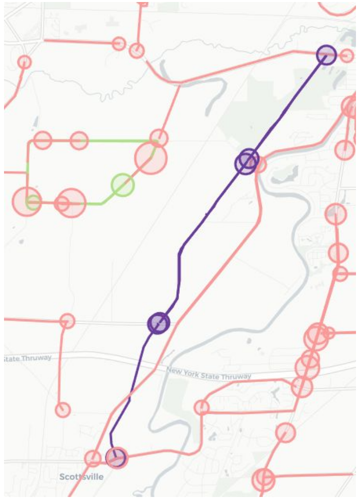
….Mountain bikers that travel from hill to hill maybe we should market here.

Small clusters

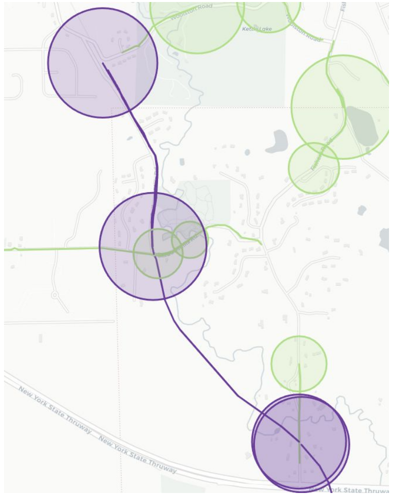
Dark purple are all smaller trails with more gravel outside of the central population.

Yellow follows most of the Auburn Trail

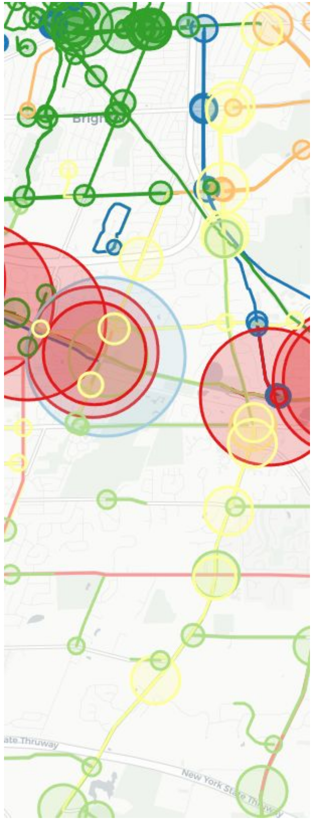
Erie Attica Trail



Genesee Valley Greenway



Railroad Mills Rd



Auburn Trail (Yellow)

Final Thoughts

By following the crisp dm methodology we are able to effectively provide a network analysis of a cool dataset.

We used 2 methods of data collection through the use of the Strava API and web scraping

Then we can reformat the data into a social network in which we can visualize and cluster the data based on the modularity algorithm.

Next we validated our clusters by mapping them along with the segment polyline to see if the clusters made sense

Finally we are able to answer our initial question by looking at the validated data and our prior knowledge to determine what the clusters represent.

https://github.com/jxv3386/StravaSNA

Sources

https://towardsdatascience.com/clustering-geospatial-data-f0584f0b04ec

https://developers.strava.com/docs/getting-started/

https://towardsdatascience.com/insights-from-visualizing-public-data-on-twitch-a73304a1b3eb