

thegraph_data_access

March 14, 2021

1 TheGraph data access

courtesy of @markusbkoch submitted by @mzargham

```
[1]: import pandas as pd
import json
import requests
import matplotlib.pyplot as plt
url = 'https://api.thegraph.com/subgraphs/name/balancer-labs/balancer'
query = ''
query {{
    pools(first: 1000, skip:{}) {{
        id
        liquidity
    }}
}}''
n = 0
pools = []
while True:
    print(f'request {n+1}')
    r = requests.post(url, json = {'query':query.format(n*1000)})
    p = json.loads(r.content)['data']['pools']
    print(f'results {len(p)}')
    pools.extend(p)
    print(f'total {len(pools)}')
    n += 1
    if len(p) < 1000:
        break
subgraph_tvl = pd.DataFrame(pools)
```

```
request 1
results 1000
total 1000
request 2
results 1000
total 2000
request 3
results 957
```

total 2957

```
[2]: subgraph_tv1
```

```
[2]:                                     id \
0      0x003a70265a3662342010823bea15dc84c6f7ed54
1      0x004e74ff81239c8f2ec0e2815defb970f3754d86
2      0x0077732357ac0f29e26ea629b79ab3b266ddb796
3      0x0092b2d25d76d84d27b999fe93d5e1c70511cd2b
4      0x0099447ef539718bba3c4d4d4b4491d307eedc53
...
2952   0xffe8c31fb0ab62c99fc6e8c724d0f1949dbaa44f
2953   0xffff293e1f6c174867f23351c1510833c8087fecb
2954   0xffff29c8bce4fbe8702e9fa16e0e6c551f364f420
2955   0xffff2a5f81d14729408201341df42af29f3b30458
2956   0xffff82910d352abe04d00d542f0ded0bfc8516f78
```

```
                                     liquidity
0      1453246.09143266209337861077499161
1      680.928486911431236447041487663722
2      0.8653140420464888814426818591183125
3      8.41728837682050716701128507145078
4      0
...
2952   2331.204011809222755546645470719059
2953   0
2954   0
2955   4370669.16065064834404384100284669
2956   0
```

[2957 rows x 2 columns]

```
[3]: subgraph_tv1.liquidity= subgraph_tv1.liquidity.apply(float)
```

```
[4]: subgraph_tv1.sort_values('liquidity', inplace=True)
```

```
[5]: subgraph_tv1.liquidity
```

```
[5]: 2956    0.000000e+00
1618    0.000000e+00
1616    0.000000e+00
2813    0.000000e+00
1613    0.000000e+00
...
1072    6.912330e+07
1614    8.310033e+07
1052    2.399595e+08
```

```
2281    3.294974e+08
363     3.344076e+08
Name: liquidity, Length: 2957, dtype: float64
```

```
[6]: plt_df=subgraph_tvl[subgraph_tvl.liquidity>1].copy().sort_values('liquidity',
↪ascending=False)
```

```
[7]: subgraph_tvl.describe()
```

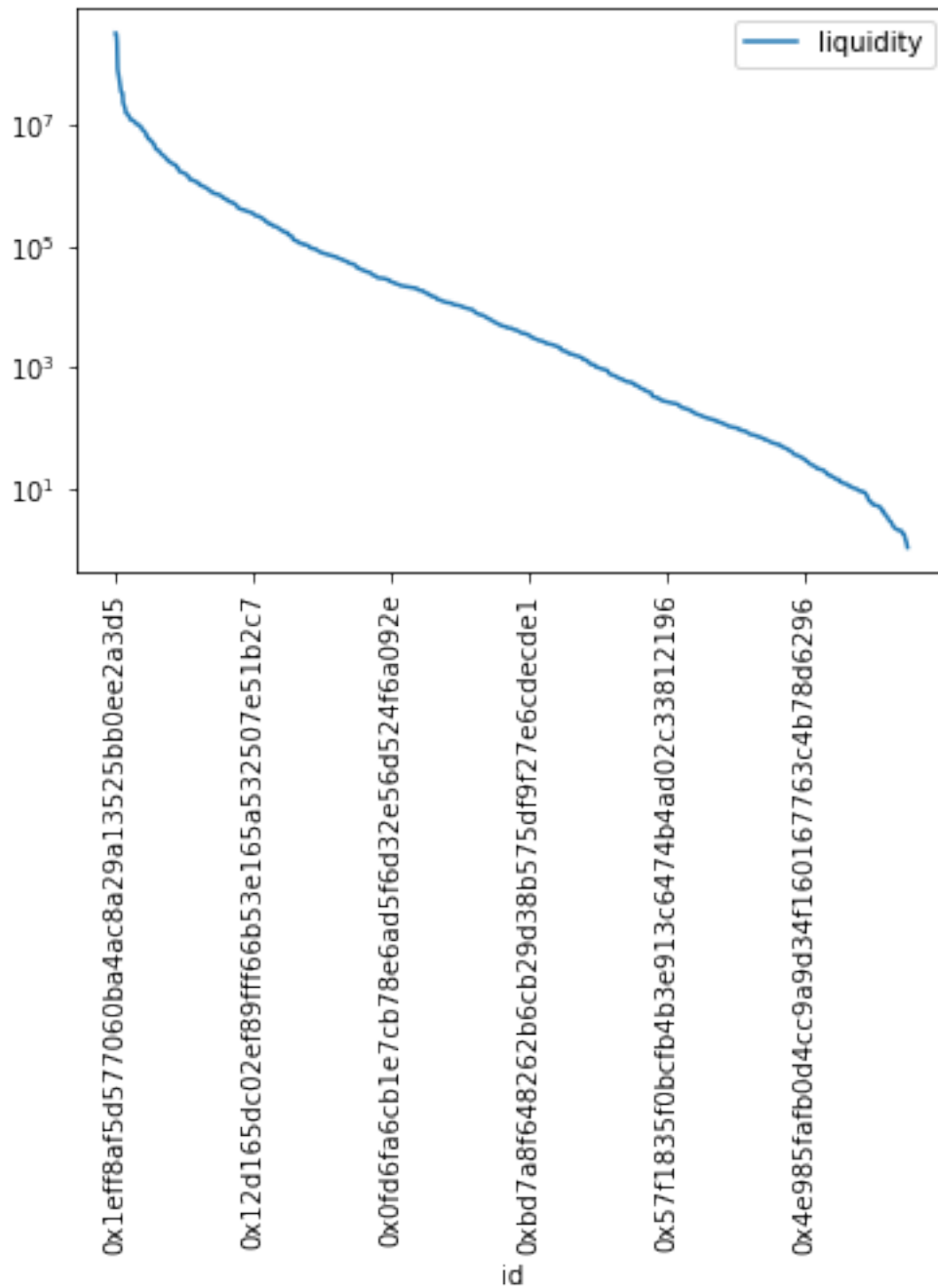
```
[7]:      liquidity
count  2.957000e+03
mean   6.839021e+05
std    1.018899e+07
min    0.000000e+00
25%    0.000000e+00
50%    0.000000e+00
75%    5.870998e+02
max     3.344076e+08
```

```
[8]: plt_df.tail()
```

```
[8]:      id liquidity
337   0x1d261ec7ab834fedb01602c5b7ffc6fc68362bbf    1.577654
981   0x53f160490d7e48ba2c31be4790f3d87a2f4dc662    1.371422
1840  0x9e4a4b53e19410ae519be74f92659e5b0ef9489b    1.330313
2344  0xcb8ec8236aff8e112517f4e9a9ffb413a237e6b7    1.153105
1289  0x6d42692518c8b09c883e7c1e69c97518107f2185    1.030083
```

```
[9]: plt_df.plot(x='id', y='liquidity', logy=True)
plt.xticks(rotation=90)
```

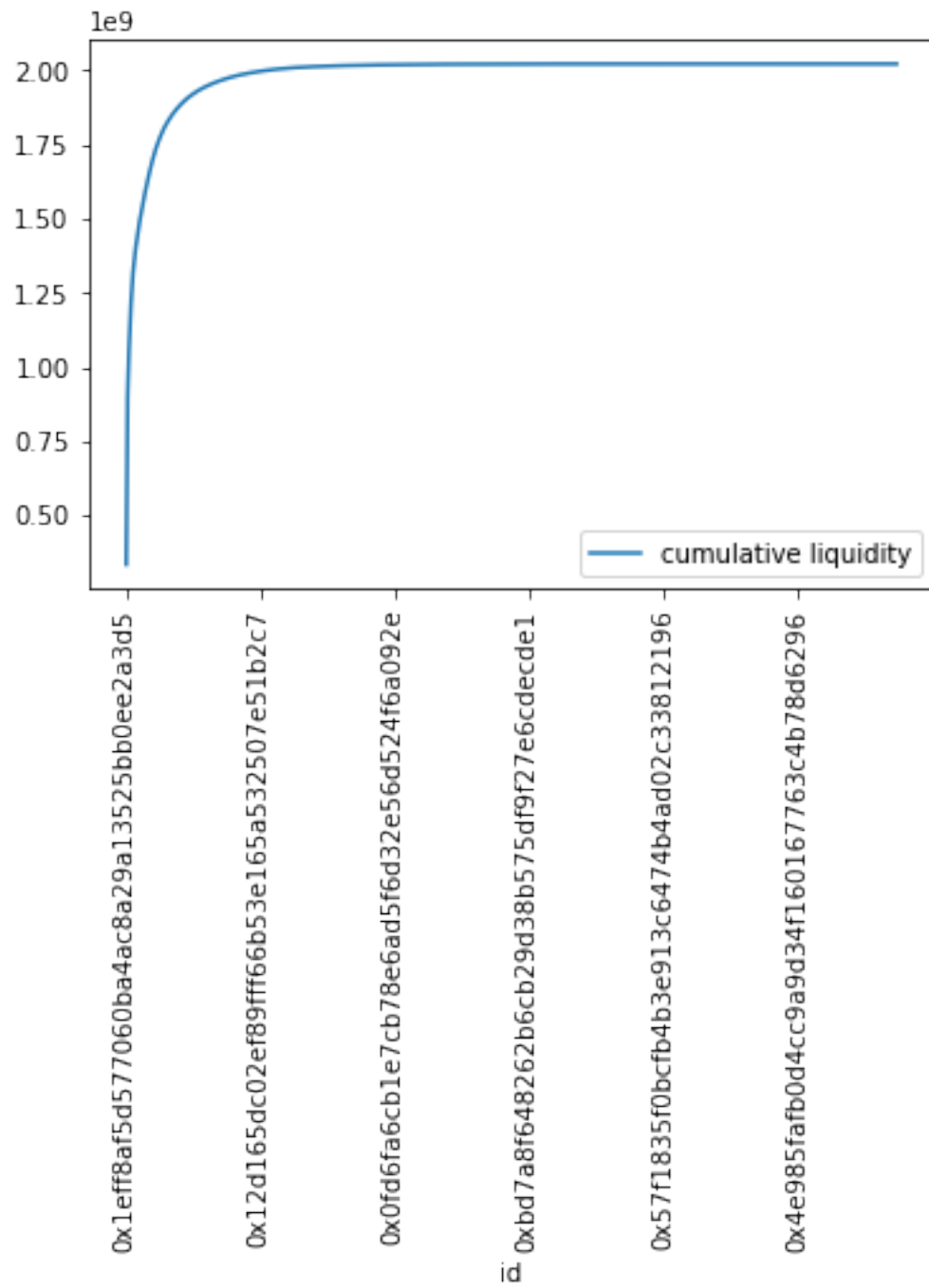
```
[9]: (array([-200.,    0.,  200.,  400.,  600.,  800., 1000., 1200., 1400.]),
[Text(-200.0, 0, '0xa8e4faa056c7f1c41f5838a896daa091dea93901'),
Text(0.0, 0, '0x1eff8af5d577060ba4ac8a29a13525bb0ee2a3d5'),
Text(200.0, 0, '0x12d165dc02ef89fff66b53e165a532507e51b2c7'),
Text(400.0, 0, '0x0fd6fa6cb1e7cb78e6ad5f6d32e56d524f6a092e'),
Text(600.0, 0, '0xbd7a8f648262b6cb29d38b575df9f27e6cdecde1'),
Text(800.0, 0, '0x57f1835f0bcfb4b3e913c6474b4ad02c33812196'),
Text(1000.0, 0, '0x4e985fafb0d4cc9a9d34f160167763c4b78d6296'),
Text(1200.0, 0, ''),
Text(1400.0, 0, '')])
```



```
[10]: plt_df['cumulative liquidity'] = plt_df.liquidity.cumsum()
```

```
[11]: plt_df.plot(x='id', y='cumulative liquidity', logy=False)
plt.xticks(rotation=90)
```

```
[11]: (array([-200.,    0.,  200.,  400.,  600.,  800., 1000., 1200., 1400.]),
      [Text(-200.0, 0, '0xa8e4faa056c7f1c41f5838a896daa091dea93901'),
       Text(0.0, 0, '0x1eff8af5d577060ba4ac8a29a13525bb0ee2a3d5'),
       Text(200.0, 0, '0x12d165dc02ef89fff66b53e165a532507e51b2c7'),
       Text(400.0, 0, '0x0fd6fa6cb1e7cb78e6ad5f6d32e56d524f6a092e'),
       Text(600.0, 0, '0xbd7a8f648262b6cb29d38b575df9f27e6cdecde1'),
       Text(800.0, 0, '0x57f1835f0bcfb4b3e913c6474b4ad02c33812196'),
       Text(1000.0, 0, '0x4e985fafb0d4cc9a9d34f160167763c4b78d6296'),
       Text(1200.0, 0, ''),
       Text(1400.0, 0, '')])
```



[]:

[]: