RELIAQUEST

An Information Security Approach to Feature Engineering

CAMLIS

Brian P. Murphy

ReliaQuest Chief Architect

RELIAQUEST

- Driving technical vision on the GreyMatter platform. Graduate of the University of Limerick in Ireland.
- Worked at Splunk, Elasticsearch, Loggly.
- ReliaQuest partners with Splunk and other Fortune 1000 customers in managing the Security Model across their entire organization.

Feature Engineering

THIS IS YOUR MACHINE LEARNING SYSTEM?
YUP! YOU POUR THE DATA INTO THIS BIG PILE OF LINEAR ALGEBRA, THEN COLLECT THE ANSWERS ON THE OTHER SIDE.
WHAT IF THE ANSWERS ARE WRONG?
JUST STIR THE PILE UNTIL THEY START LOOKING RIGHT.
DATA
ANSWERS

Feature Engineering

Definition

- Feature engineering is the process of using domain knowledge of the data to create features that make machine learning algorithms work.

- “Coming up with features is difficult, time-consuming, requires expert knowledge. ‘Applied machine learning’ is basically feature engineering.”
  — Andrew Ng, Machine Learning and AI via Brain simulations

Feature Engineering

RELIAQUEST

Outcome Desired

- Create encoders that can represent individual elements in a set of related values while also maintaining their relationships.

- URIs / Domains / Log messages
  - Classify and detect outlier and/or malicious activity

- Geo locations
  - Typically similar attacks will come from the same geo regions, but not necessarily the same countries.

Feature Engineering

Machine Learning

- Most algorithms require columnar numerical values
- Models suffer from GIGO (Garbage In/Garbage Out)
- Memory constraints normally impossible to hold the entire data set in RAM

Feature Engineering

Common Options

- Feature hashing
- One hot encoding
- Ordinal/Label encoding

Feature Engineering

Issues

- Feature hashing
  - Locality of information is lost

- One hot encoding
  - Can result in feature explosion
  - Dealing with new values not seen in training set

- Ordinal/Label encoding
  - Locality is not always obvious
  - Dealing with new values is hard also

Feature Engineering

Proposed solution

- Min Hash Shingle
- Min Hash ngram
- Geo Hash

Feature Engineering

ngram/shingling

- Ngrams are contiguous sequences of characters from events.
  - Camlis -> 3gram -> [“cam”,”aml”,”mli”,”lis”]

- Shingles are to tokens what ngrams are to characters.
  - Camlis is really awesome -> 3shingle -> [“camlisisreally”, ”isreallyawesome”]

Feature Engineering

Min Hashing

- Jaccard Similarity
  - Very accurate.
  - But need to compare every element of a set with every other element of the set.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}.$$

- Min Hash
  - Approximates Jaccard by creating ‘k’ hash functions and hashing each ngram or shingle, then finding the min value for each hash.
  - Events that share min hashes are similar.

Vassilvitskii, Sergey - COMS 6998-12

Feature Engineering

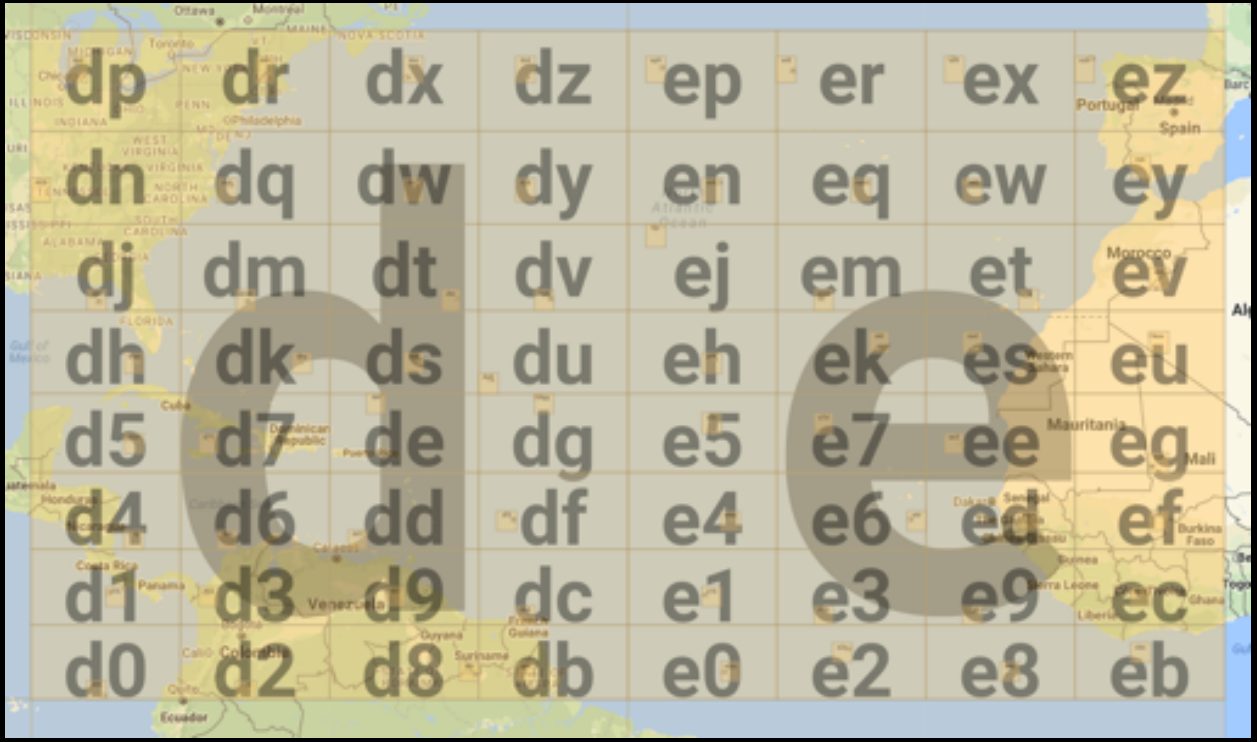
Min Hash Benefits

- Encodes all values into a known number of columns.
- New values are handled and if they are similar to fitted values they will be similarly encoded.
- Linear time event by event processing.

Feature Engineering

Geo Hash Approach

- Partitions the globe into a hierarchical NxN (32x32) grid, using a z-curve.



Feature Engineering

Geo Hash Approach

RELIAQUEST

- Calculate a bivariate (3D) normal distribution with the location at the peak of this distribution.

- Based on distance from the grid peak assign values across the grid.

- Results in NxN columns representing a decaying weight from the detected action.

Feature Engineering

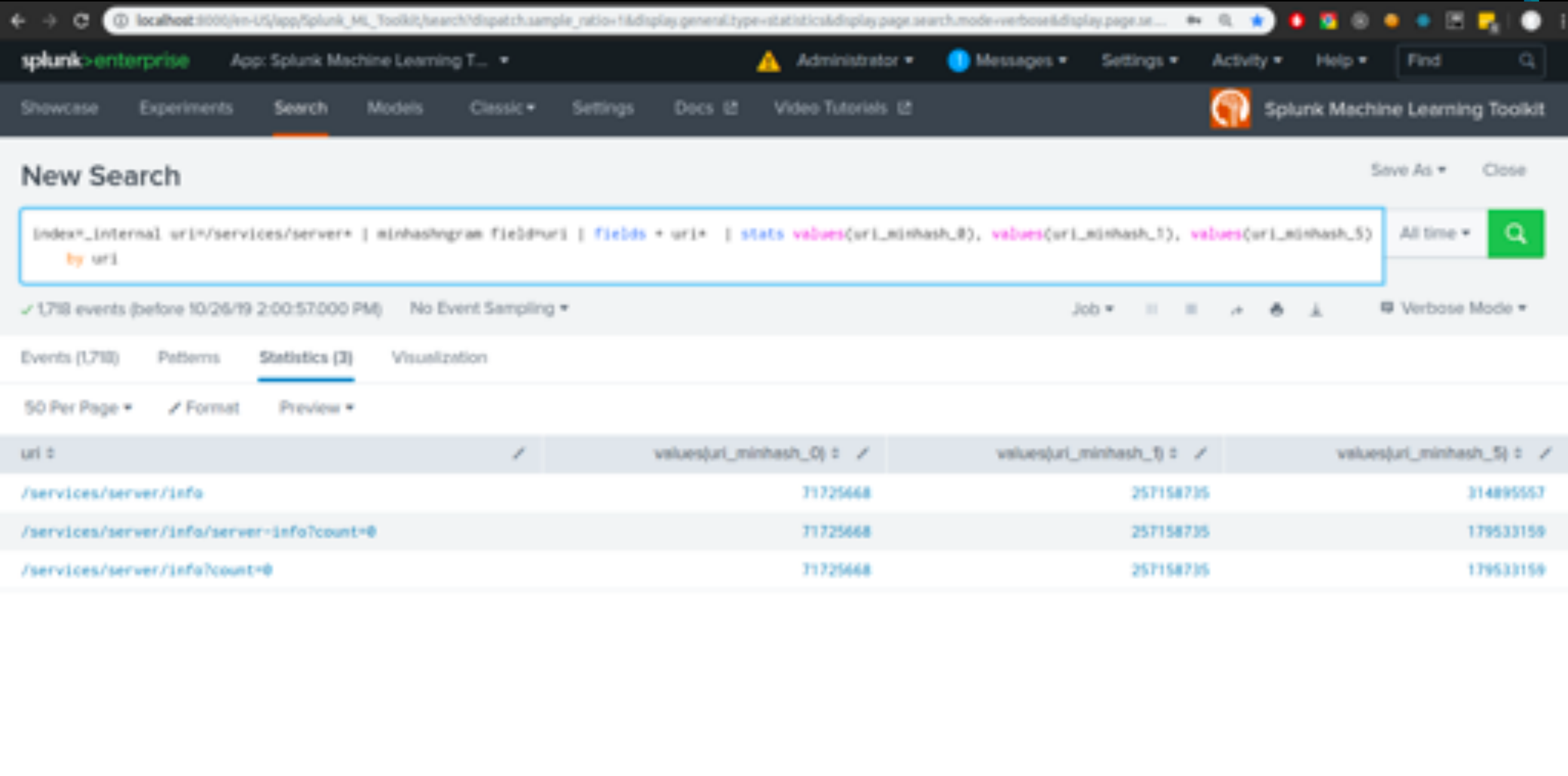
Geo Hash Benefits

- Allows the encoding of all possible geolocations into known number of columns.
- New locations are allowed for in the encoding.
- Maintains relationships between geolocations.
- Lookups and values can be precomputed to accelerate encoding.

Feature Engineering

Demo

Do we get to mess with him now?
No. We wait until the worst possible moment.
ML
Demo

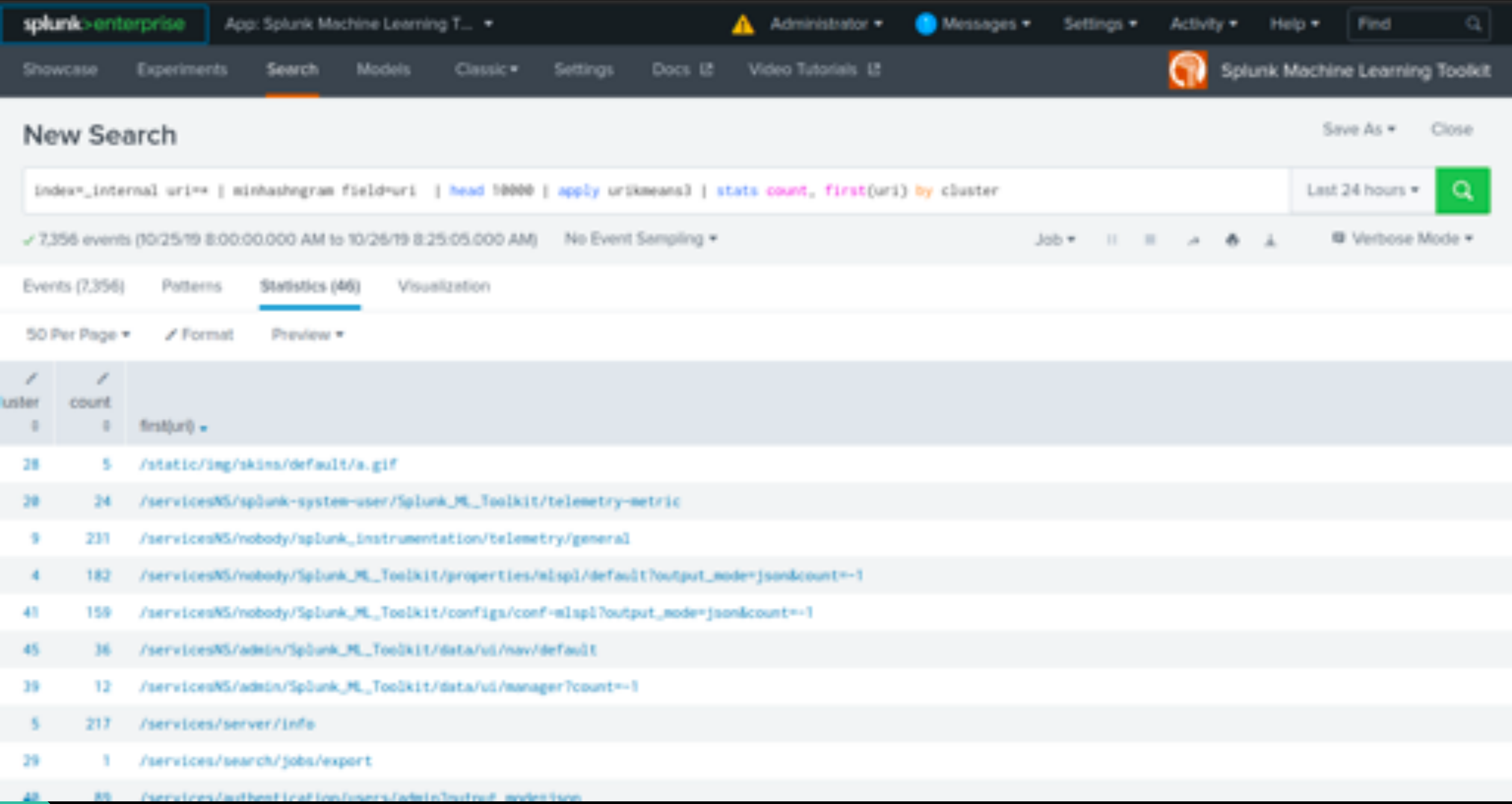


New Search

```
index=_internal uri=/services/server* | minhashngram field=uri | fields + uri*  | stats values(uri_minhash_0), values(uri_minhash_1), values(uri_minhash_5)
    by uri
```

✓ 1,718 events (before 10/26/19 2:00:57.000 PM) No Event Sampling

uri	values(uri_minhash_0)	values(uri_minhash_1)	values(uri_minhash_5)
/services/server/info	71725668	257158735	314895557
/services/server/info/server-info?count=0	71725668	257158735	179533159
/services/server/info?count=0	71725668	257158735	179533159



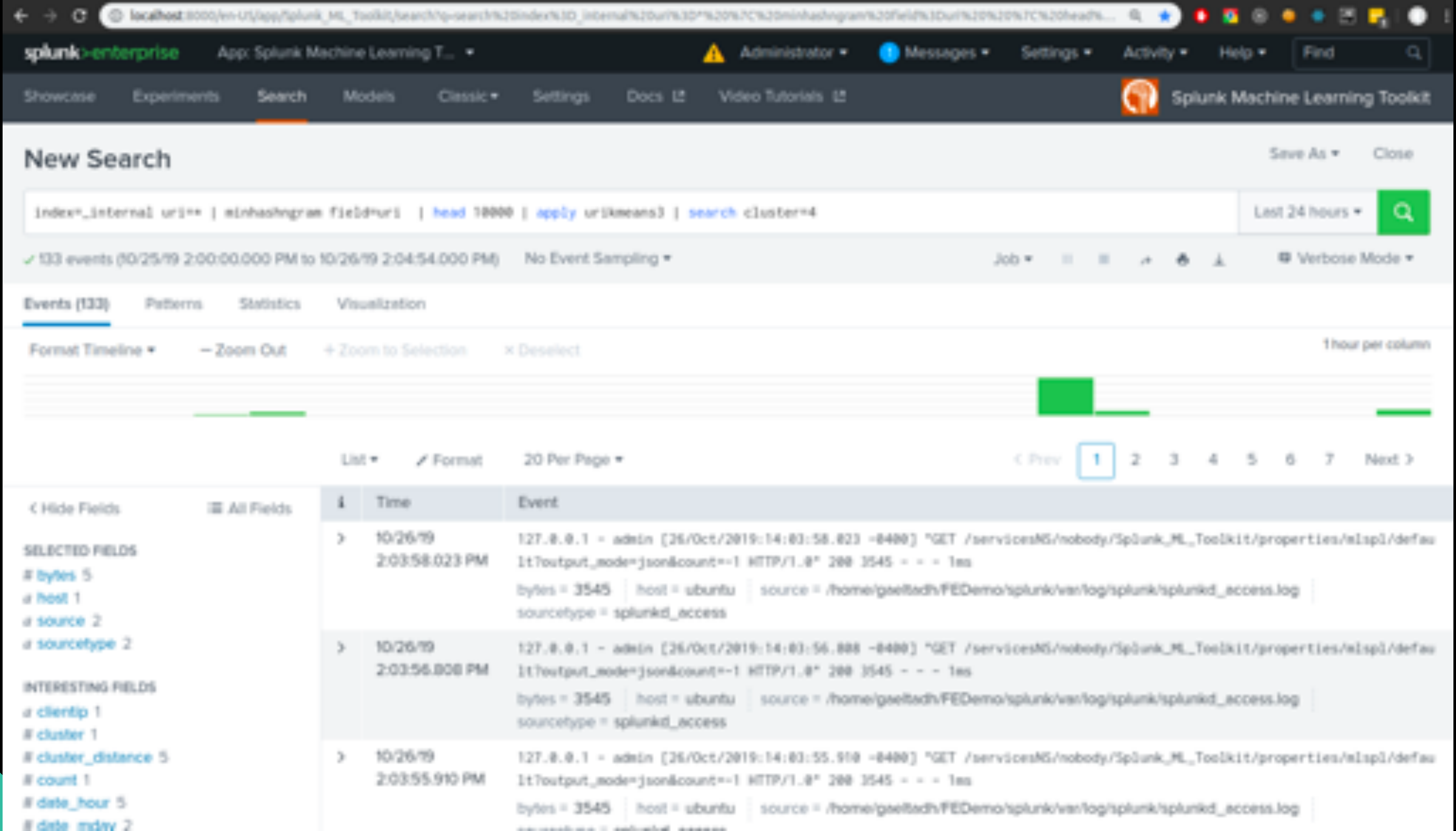
New Search

```
index=_internal uri=* | minhashngram field=uri  | head 10000 | apply urikmeans3 | stats count, first(uri) by cluster
```

✓ 7,356 events (10/25/19 8:00:00.000 AM to 10/26/19 8:25:05.000 AM) No Event Sampling

Events (7,356) Patterns Statistics (46) Visualization

uster	count	first(uri)
28	5	/static/img/skins/default/a.gif
20	24	/servicesNS/splunk-system-user/Splunk_ML_Toolkit/telemetry-metric
9	231	/servicesNS/nobody/splunk_instrumentation/telemetry/general
4	182	/servicesNS/nobody/Splunk_ML_Toolkit/properties/mlspl/default?output_mode=json&count=-1
41	159	/servicesNS/nobody/Splunk_ML_Toolkit/configs/conf-mlspl?output_mode=json&count=-1
45	36	/servicesNS/admin/Splunk_ML_Toolkit/data/ui/nav/default
39	12	/servicesNS/admin/Splunk_ML_Toolkit/data/ui/manager?count=-1
5	217	/services/server/info
29	1	/services/search/jobs/export
40	89	/services/authentication/users/admin?output_mode=json



New Search

```
index=_internal uri=* | minhashngram field=uri  | head 10000 | apply urikmeans3 | search cluster=4
```

✓ 133 events (10/25/19 2:00:00.000 PM to 10/26/19 2:04:54.000 PM)

SELECTED FIELDS
- # bytes 5
- a host 1
- a source 2
- a sourcetype 2

INTERESTING FIELDS
- a clientip 1
- # cluster 1
- # cluster_distance 5
- # count 1
- # date_hour 5
- # date_mday 2

10/26/19 2:03:58.023 PM

```
127.0.0.1 - admin [26/Oct/2019:14:03:58.023 -0400] "GET /servicesNS/nobody/Splunk_ML_Toolkit/properties/mlspl/defau
lt?output_mode=json&count=-1 HTTP/1.0" 200 3545 - - - 1ms
```

bytes = 3545 | host = ubuntu | source = /home/gaeltadh/FEDemo/splunk/var/log/splunk/splunkd_access.log | sourcetype = splunkd_access

10/26/19 2:03:56.808 PM

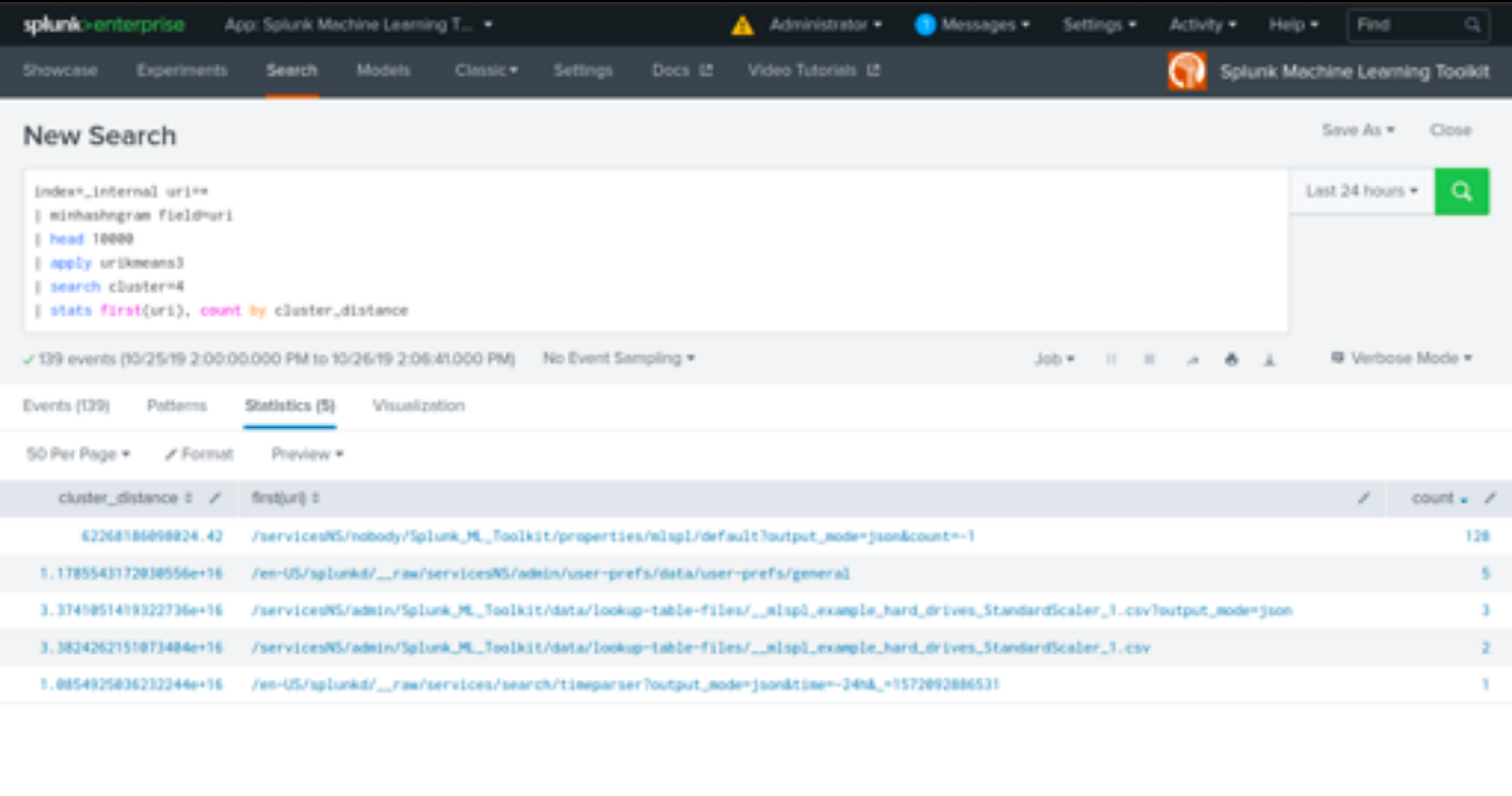
```
127.0.0.1 - admin [26/Oct/2019:14:03:56.808 -0400] "GET /servicesNS/nobody/Splunk_ML_Toolkit/properties/mlspl/defau
lt?output_mode=json&count=-1 HTTP/1.0" 200 3545 - - - 1ms
```

bytes = 3545 | host = ubuntu | source = /home/gaeltadh/FEDemo/splunk/var/log/splunk/splunkd_access.log | sourcetype = splunkd_access

10/26/19 2:03:55.910 PM

```
127.0.0.1 - admin [26/Oct/2019:14:03:55.910 -0400] "GET /servicesNS/nobody/Splunk_ML_Toolkit/properties/mlspl/defau
lt?output_mode=json&count=-1 HTTP/1.0" 200 3545 - - - 1ms
```

bytes = 3545 | host = ubuntu | source = /home/gaeltadh/FEDemo/splunk/var/log/splunk/splunkd_access.log

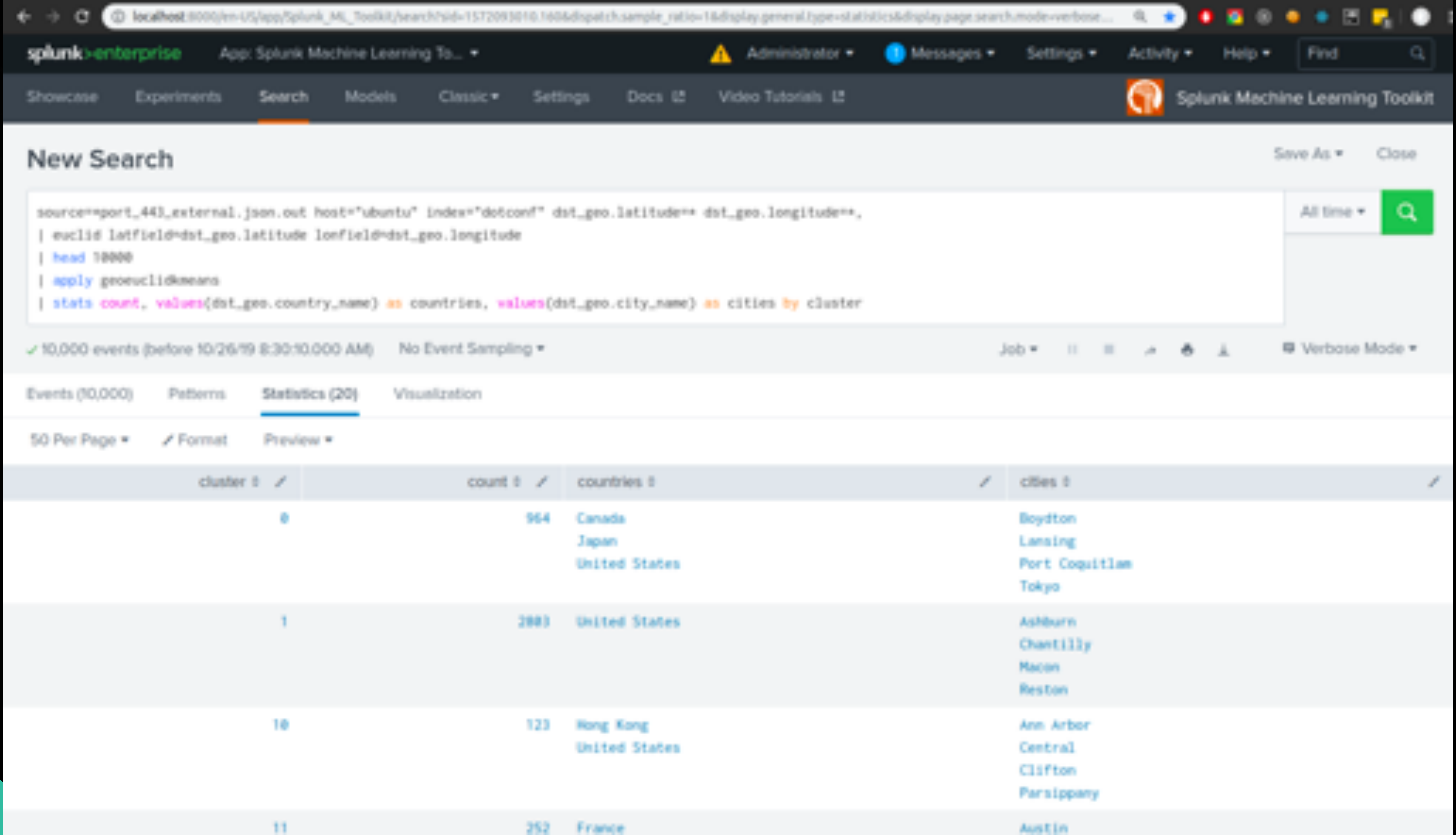


New Search

```
index=_internal uri=*
| minhashngram field=uri
| head 10000
| apply urikmeans3
| search cluster=4
| stats first(uri), count by cluster_distance
```

✓ 139 events (10/25/19 2:00:00.000 PM to 10/26/19 2:06:41.000 PM)   No Event Sampling

cluster_distance	first(uri)	count
62268186098024.42	/servicesNS/nobody/Splunk_ML_Toolkit/properties/mlspl/default?output_mode=json&count=-1	128
1.1785543172030556e+16	/en-US/splunkd/__raw/servicesNS/admin/user-prefs/data/user-prefs/general	5
3.3741051419322736e+16	/servicesNS/admin/Splunk_ML_Toolkit/data/lookup-table-files/__mlspl_example_hard_drives_StandardScaler_1.csv?output_mode=json	3
3.3824262151073404e+16	/servicesNS/admin/Splunk_ML_Toolkit/data/lookup-table-files/__mlspl_example_hard_drives_StandardScaler_1.csv	2
1.0854925036232244e+16	/en-US/splunkd/__raw/services/search/timeparser?output_mode=json&time=-24h&_=1572092886531	1

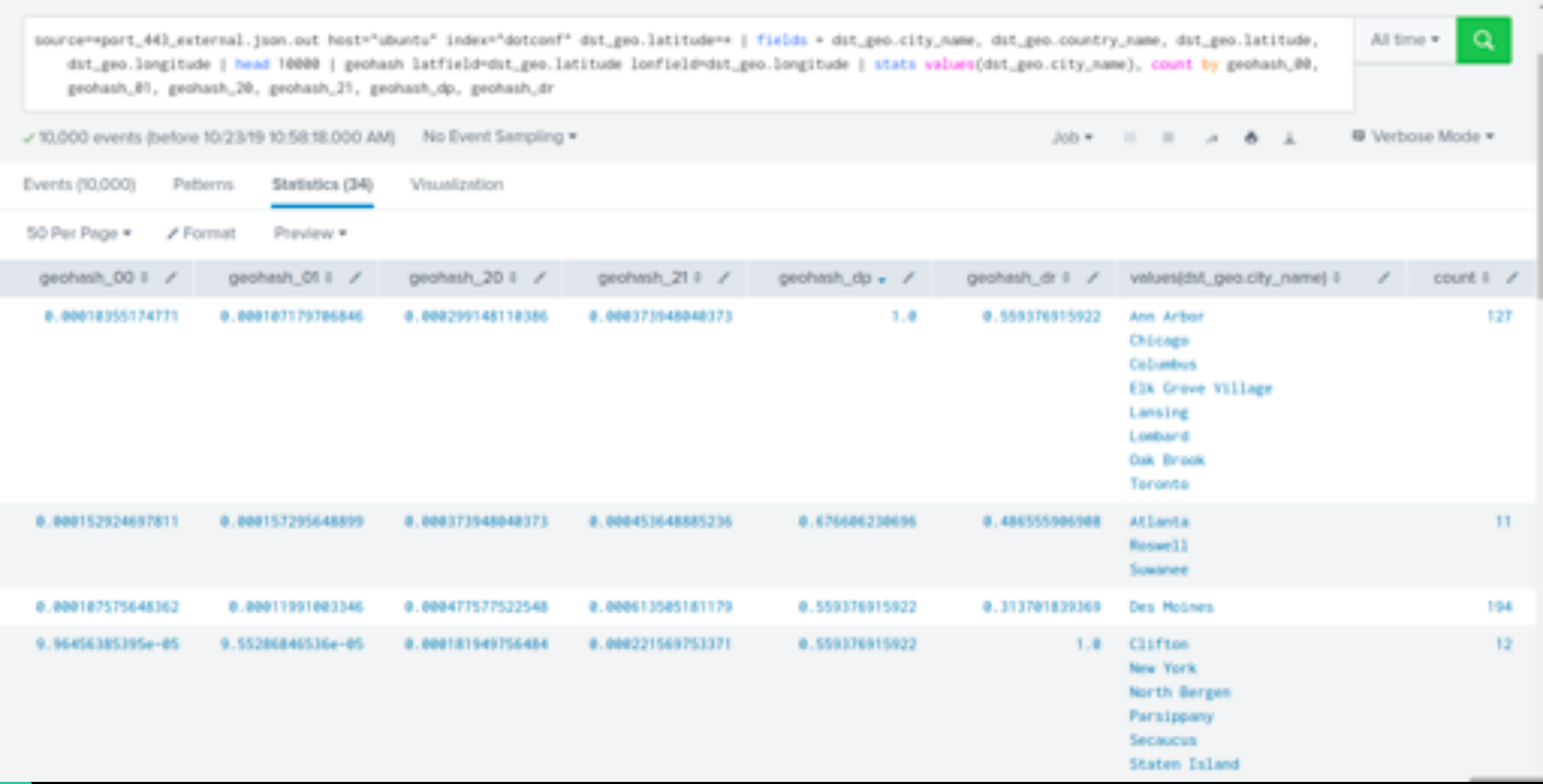


New Search

```
source=port_443_external.json.out host="ubuntu" index="dotconf" dst_geo.latitude=* dst_geo.longitude=*,
| euclid latfield=dst_geo.latitude lonfield=dst_geo.longitude
| head 10000
| apply geoeuclidkmeans
| stats count, values(dst_geo.country_name) as countries, values(dst_geo.city_name) as cities by cluster
```

✓ 10,000 events (before 10/26/19 8:30:10.000 AM) No Event Sampling

cluster	count	countries	cities
0	964	Canada Japan United States	Boydton Lansing Port Coquitlam Tokyo
1	2003	United States	Ashburn Chantilly Macon Reston
10	123	Hong Kong United States	Ann Arbor Central Clifton Parsippany
11	252	France	Austin

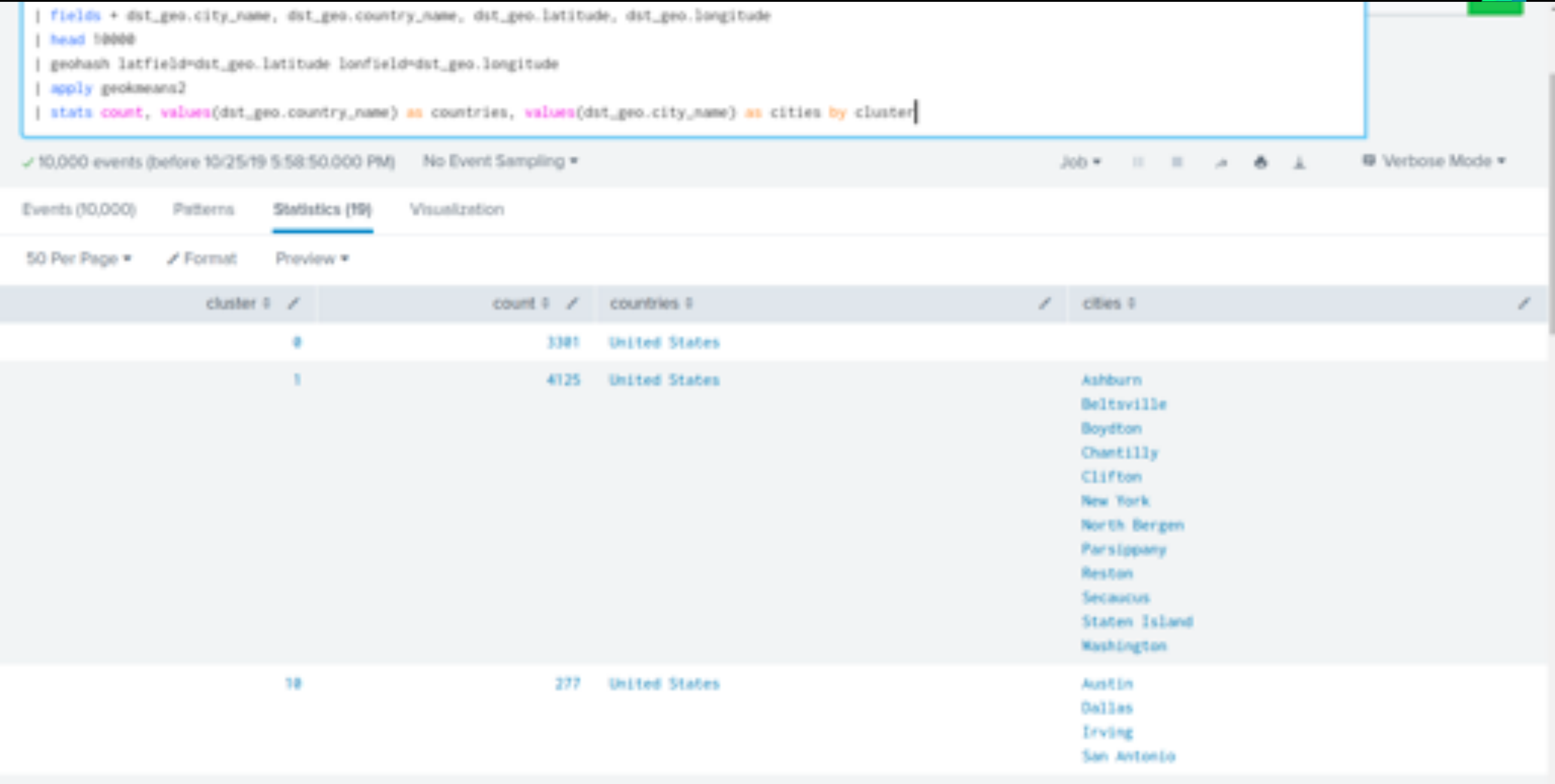


```
source=*port_443_external.json.out host="ubuntu" index="dotconf" dst_geo.latitude=* | fields + dst_geo.city_name, dst_geo.country_name, dst_geo.latitude,
    dst_geo.longitude | head 10000 | geohash latfield=dst_geo.latitude lonfield=dst_geo.longitude | stats values(dst_geo.city_name), count by geohash_00,
    geohash_01, geohash_20, geohash_21, geohash_dp, geohash_dr
```

✓ 10,000 events (before 10/23/19 10:58:18.000 AM)

No Event Sampling ▾

geohash_00	geohash_01	geohash_20	geohash_21	geohash_dp	geohash_dr	values(dst_geo.city_name)	count
0.00018355174771	0.000107179706846	0.000299148118386	0.000373948048373	1.0	0.559376915922	Ann Arbor Chicago Columbus Elk Grove Village Lansing Lombard Oak Brook Toronto	127
0.000152924697811	0.000157295648899	0.000373948048373	0.000453648885236	0.676606230696	0.486555906908	Atlanta Roswell Suwanee	11
0.000187575648362	0.00011991003346	0.000477577522548	0.000613585181179	0.559376915922	0.313701839369	Des Moines	194
9.96456385395e-05	9.55286846536e-05	0.000181949756484	0.000221569753371	0.559376915922	1.0	Clifton New York North Bergen Parsippany Secaucus Staten Island	12



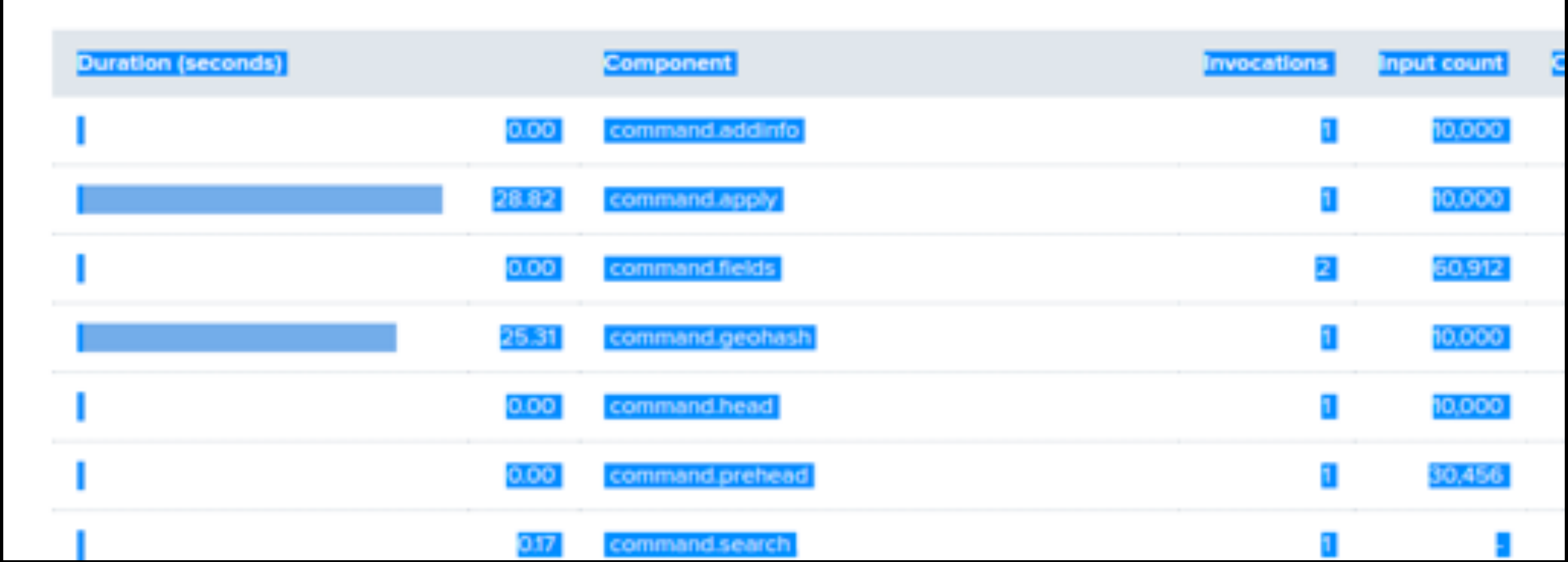
```
| fields + dst_geo.city_name, dst_geo.country_name, dst_geo.latitude, dst_geo.longitude
| head 10000
| geohash latfield=dst_geo.latitude lonfield=dst_geo.longitude
| apply geokmeans2
| stats count, values(dst_geo.country_name) as countries, values(dst_geo.city_name) as cities by cluster
```

✓ 10,000 events (before 10/25/19 5:58:50.000 PM) No Event Sampling ▾

cluster	count	countries	cities
0	3301	United States	
1	4125	United States	Ashburn Beltsville Boydton Chantilly Clifton New York North Bergen Parsippany Reston Secaucus Staten Island Washington
10	277	United States	Austin Dallas Irving San Antonio

Feature Engineering

Demo



Duration (seconds)	Component	Invocations	Input count
0.00	command.addinfo	1	10,000
28.82	command.apply	1	10,000
0.00	command.fields	2	50,912
25.31	command.geohash	1	10,000
0.00	command.head	1	10,000
0.00	command.prehead	1	30,456
0.17	command.search	1	-

Thank You

https://github.com/GaelTadh/rq_feature_engineering
(coming soon)

pmurphy@reliaquest.com