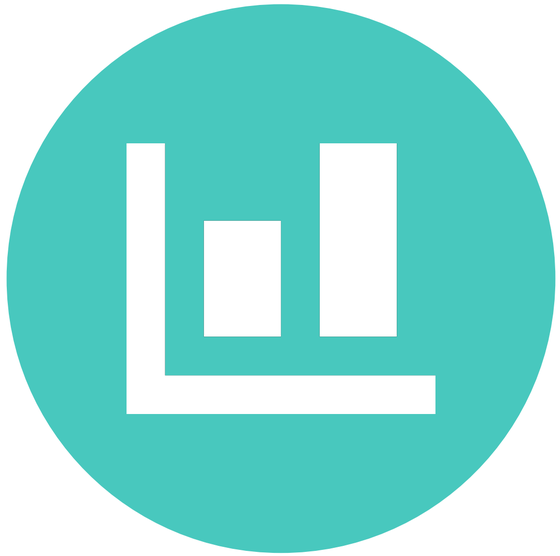
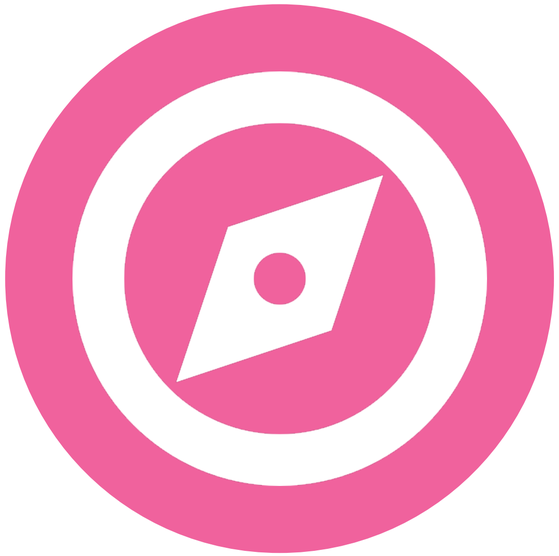
Kibana 5

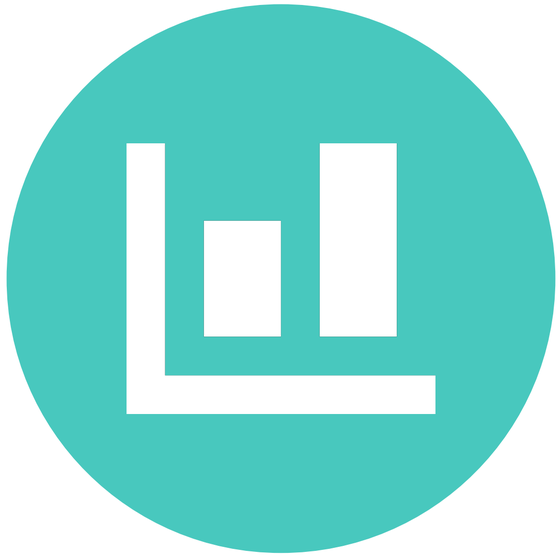
Rashid Khan **Kibana Creator** @rashidkpc

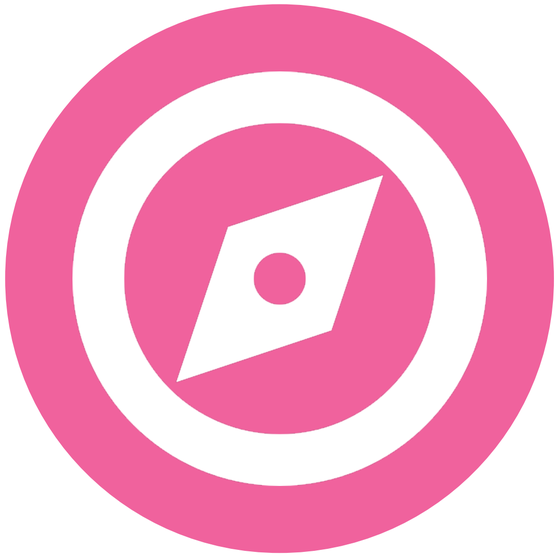
elastic





elastic

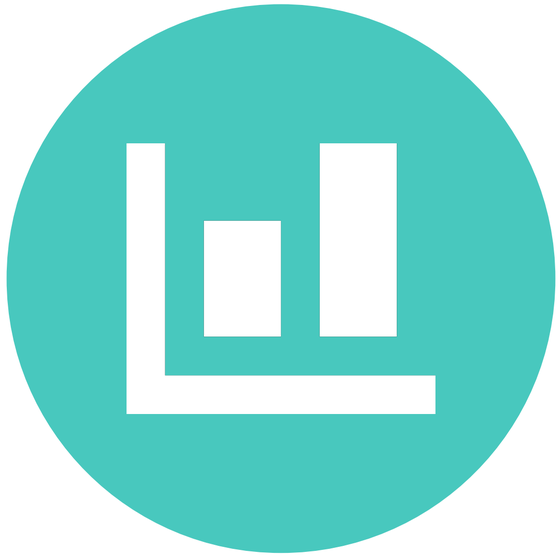


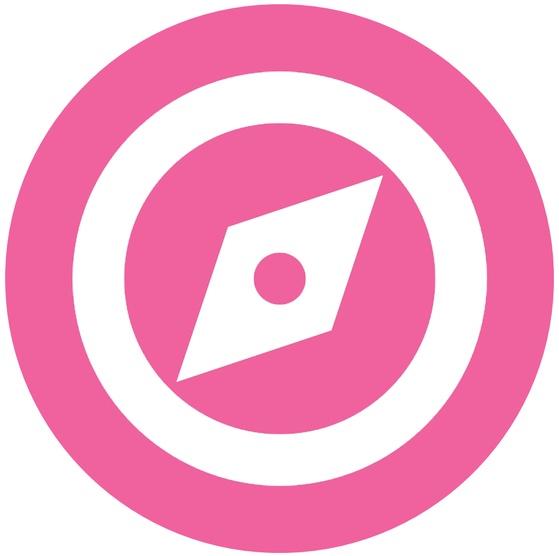


discover

elastic

discover

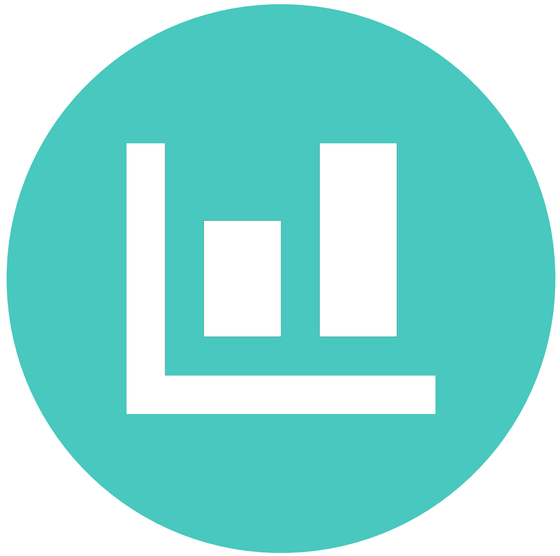




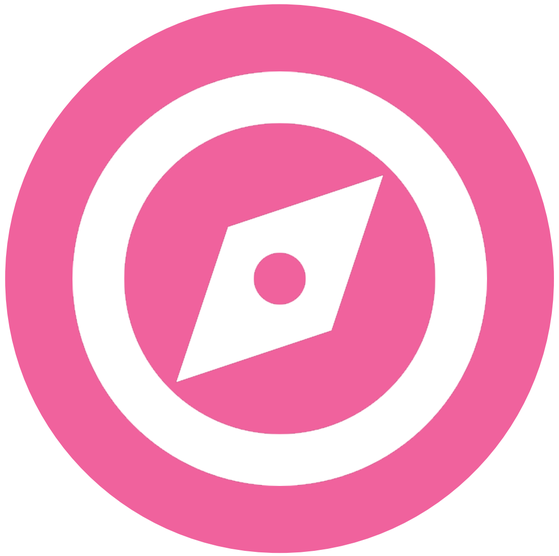
dashboard

elastic

discover



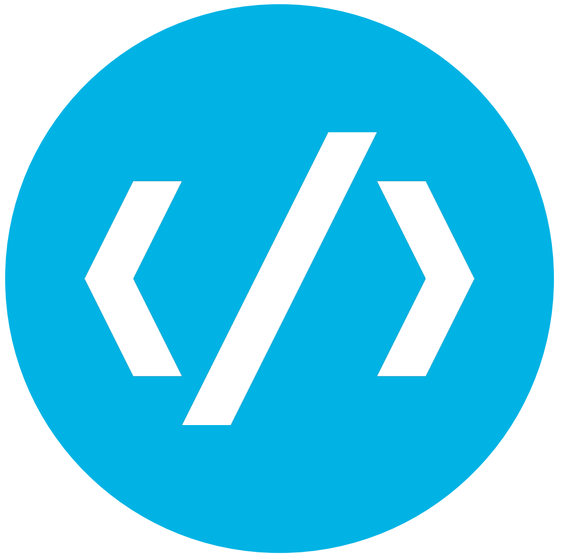
visualize



dashboard

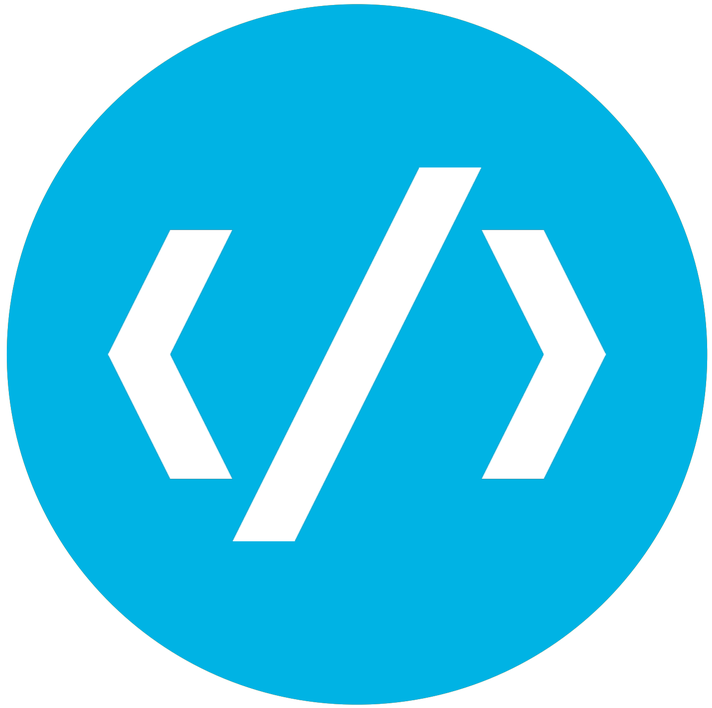
elastic

elastic

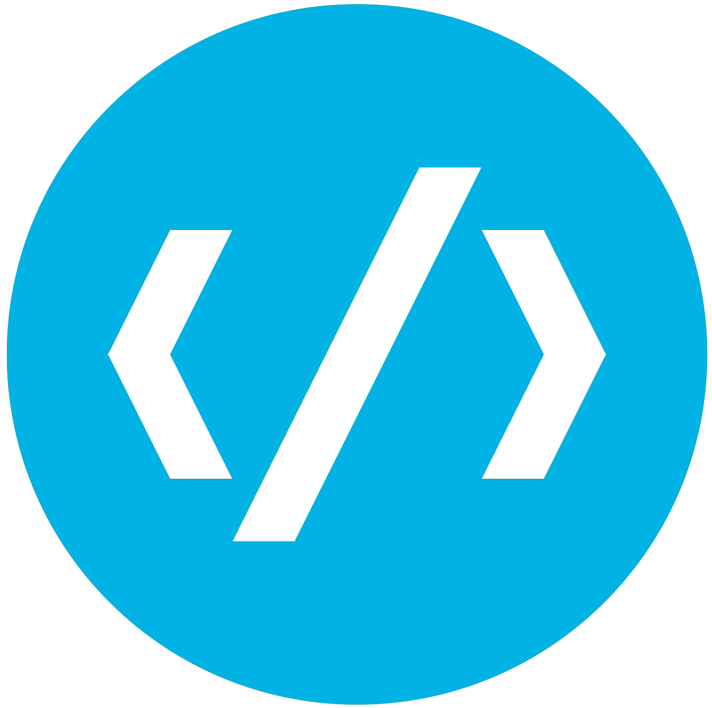


console

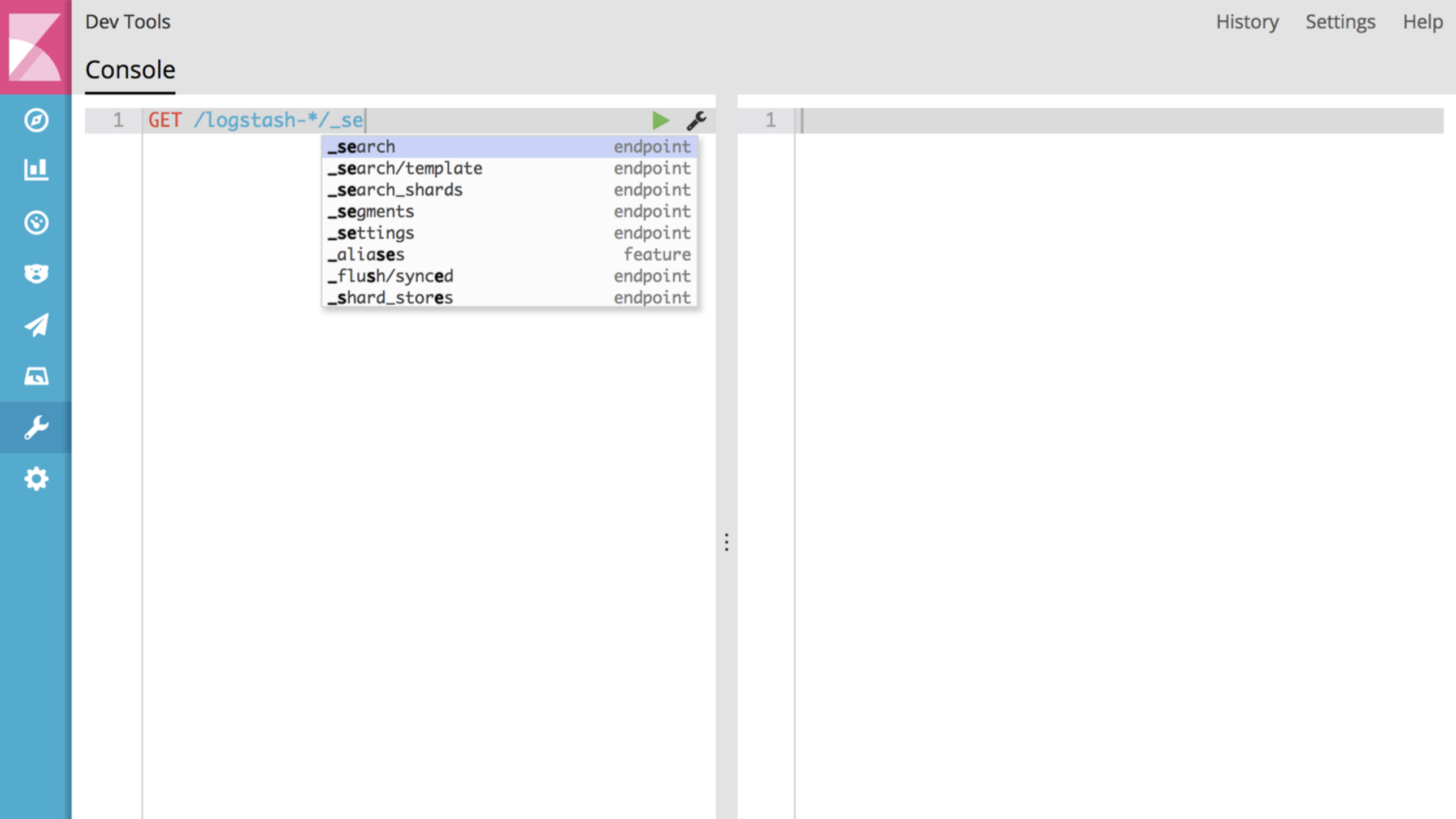
elastic

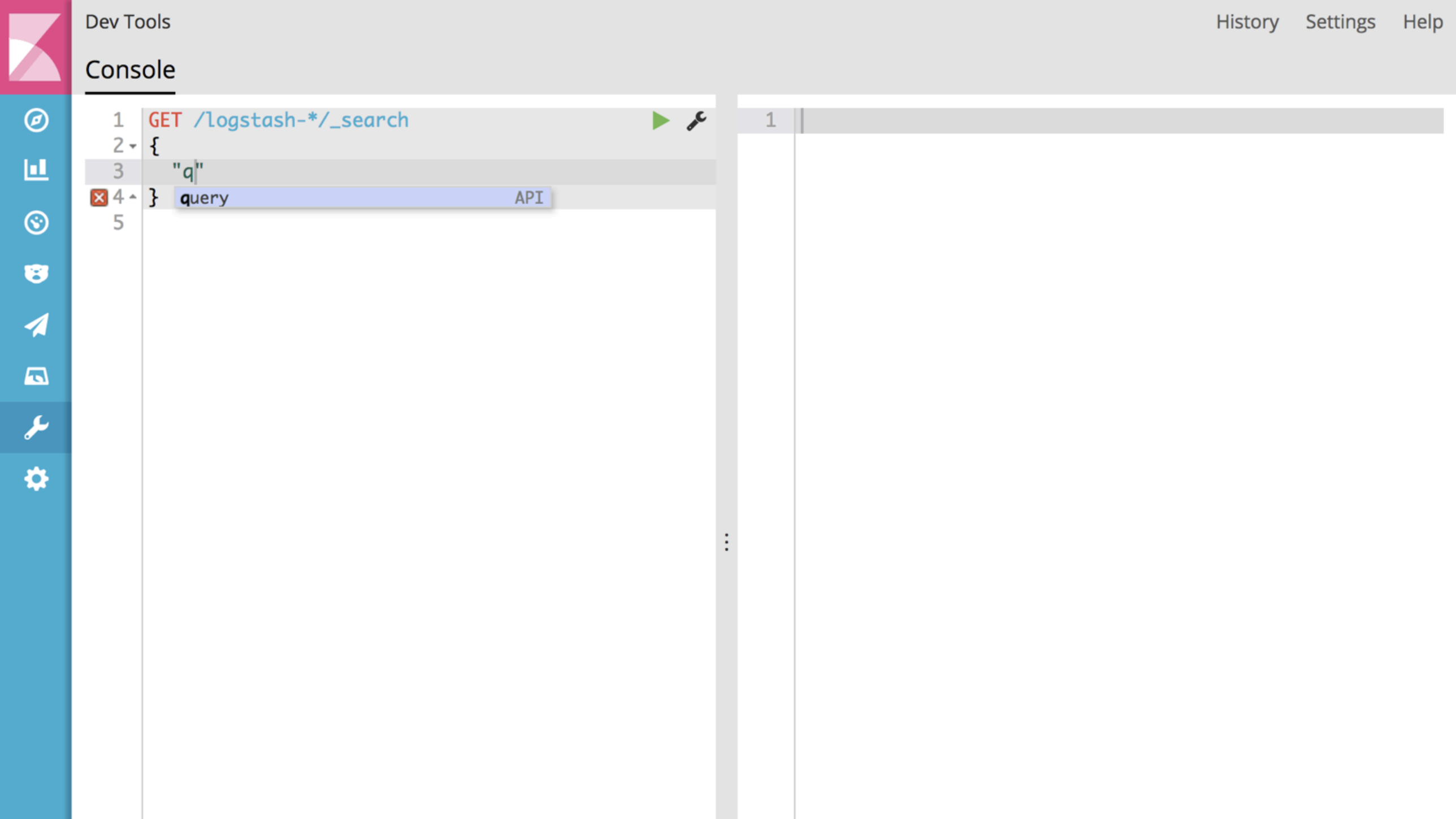


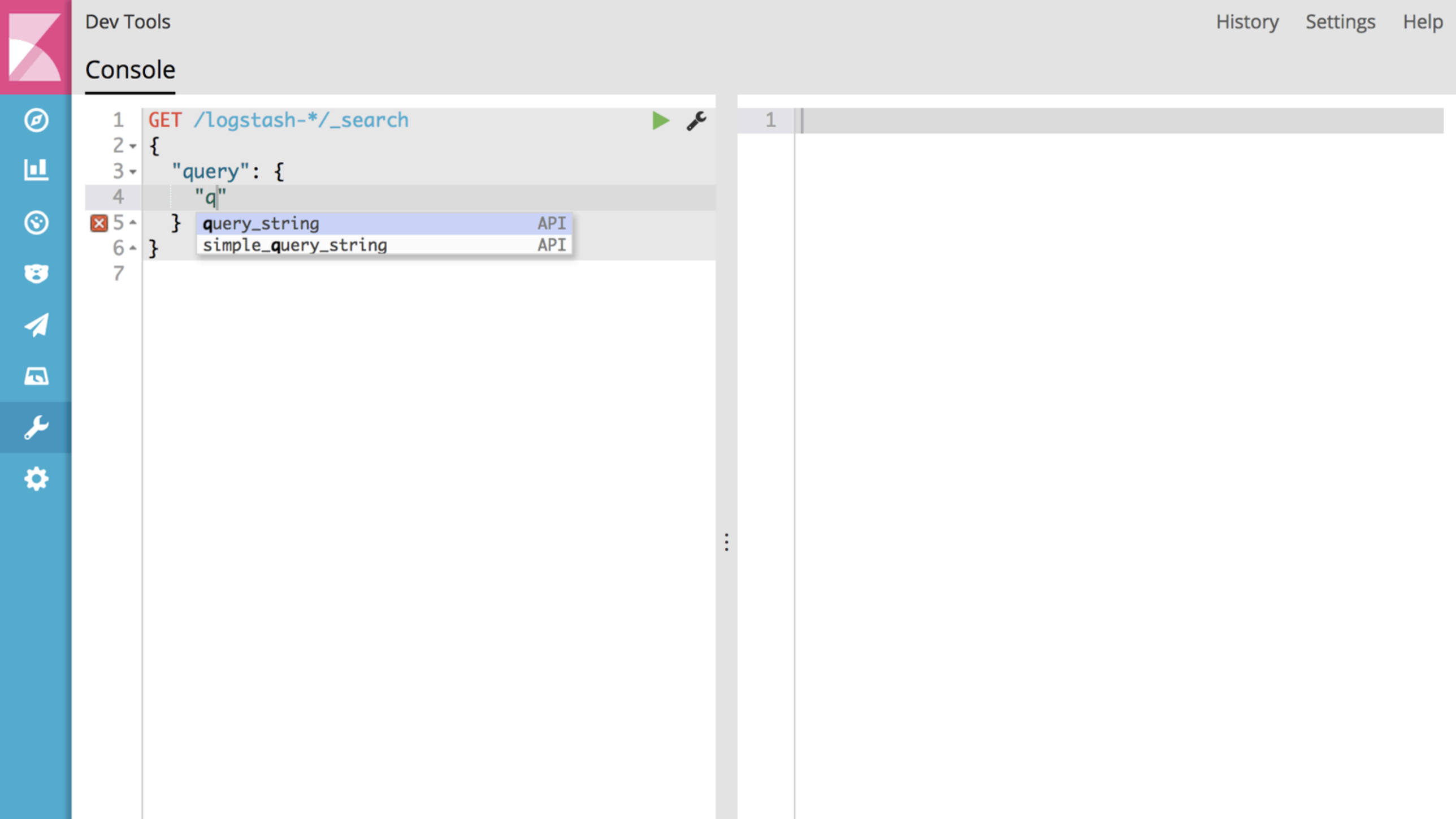
console

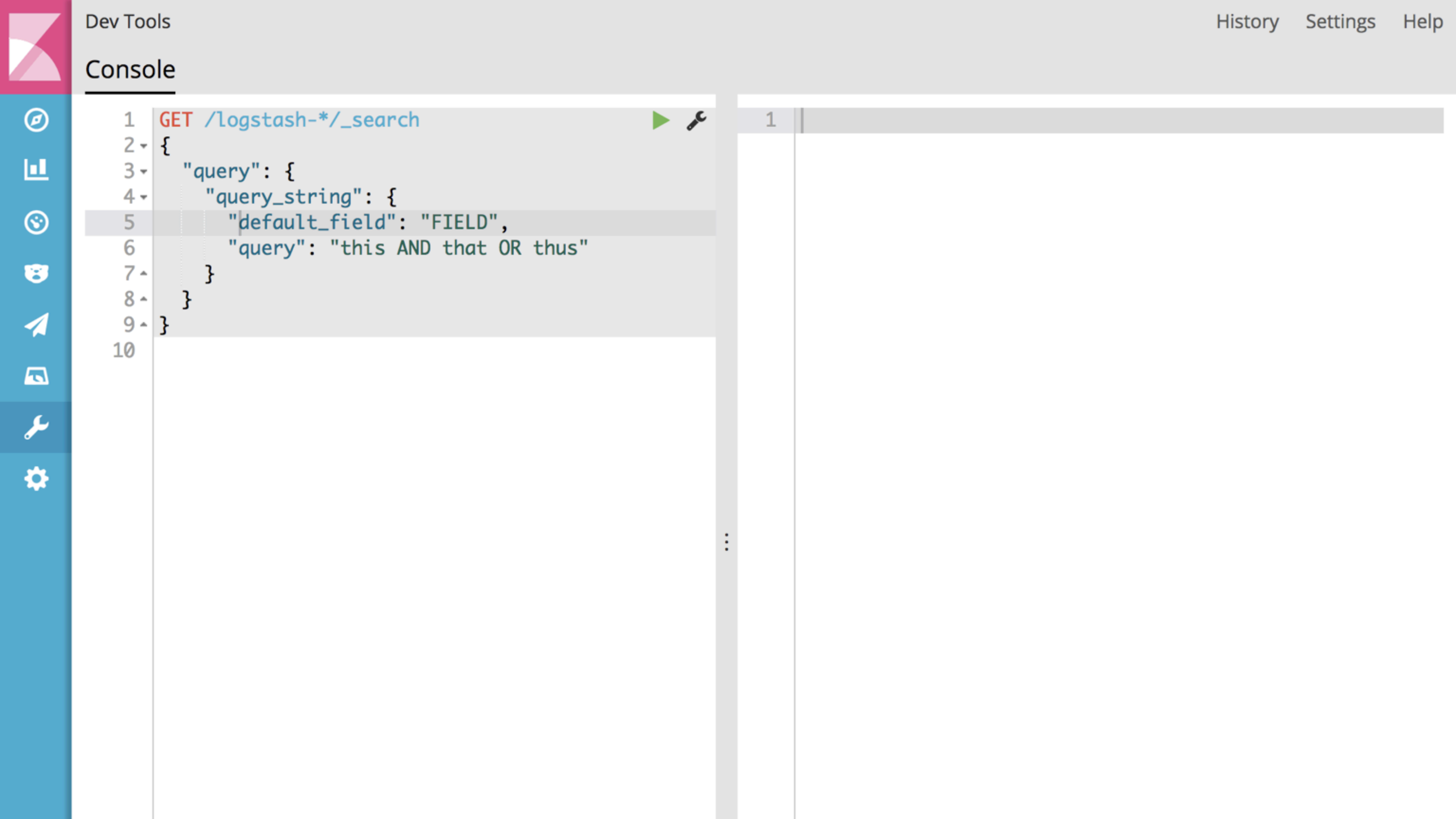


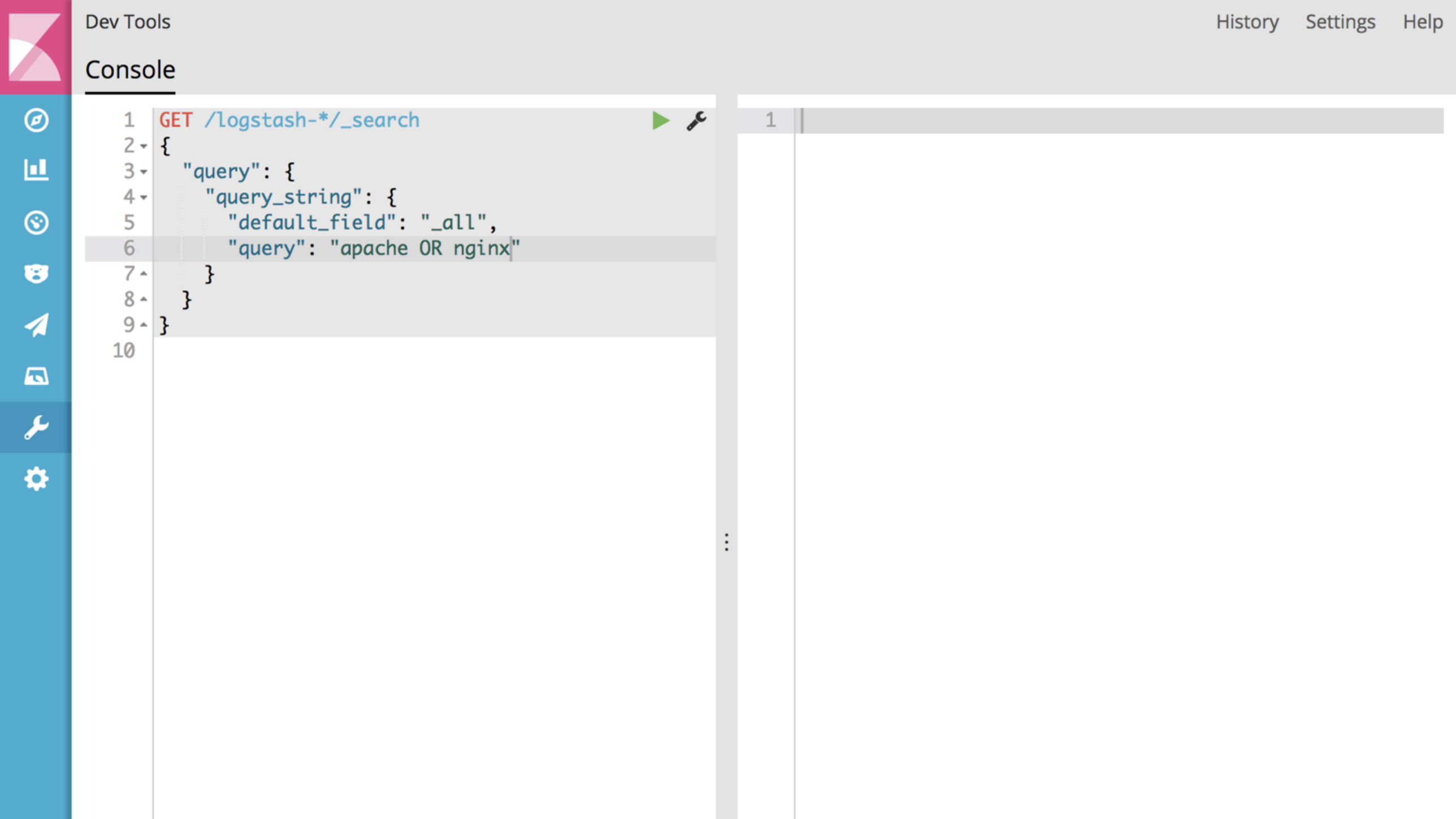
{“total”:”control”}

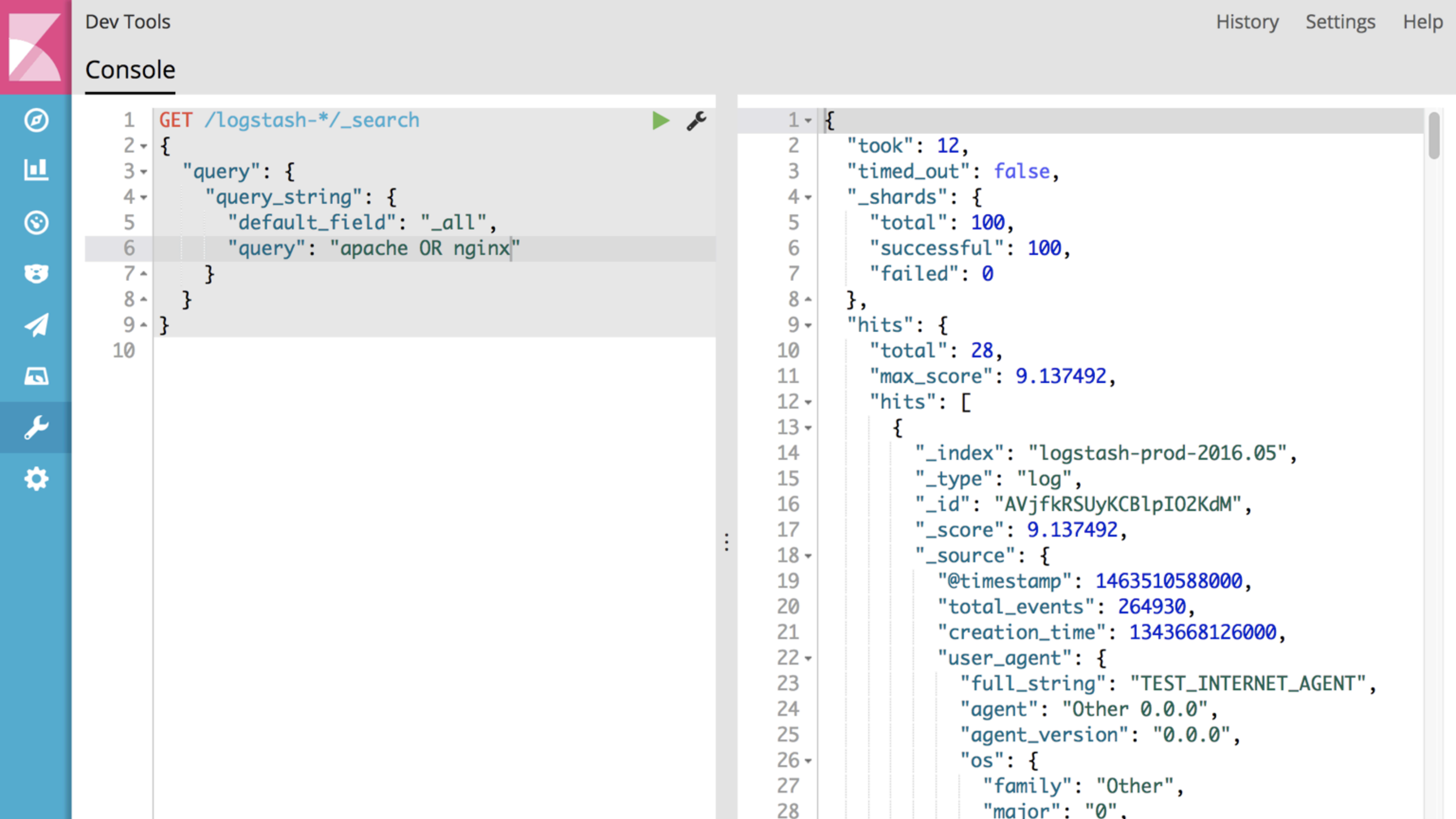

Dev Tools
History
Settings
Help
Console
1
GET /logstash-*/_se
_search endpoint
_search/template endpoint
_search_shards endpoint
_segments endpoint
_settings endpoint
_aliases feature
_flush/synced endpoint
_shard_stores endpoint
1

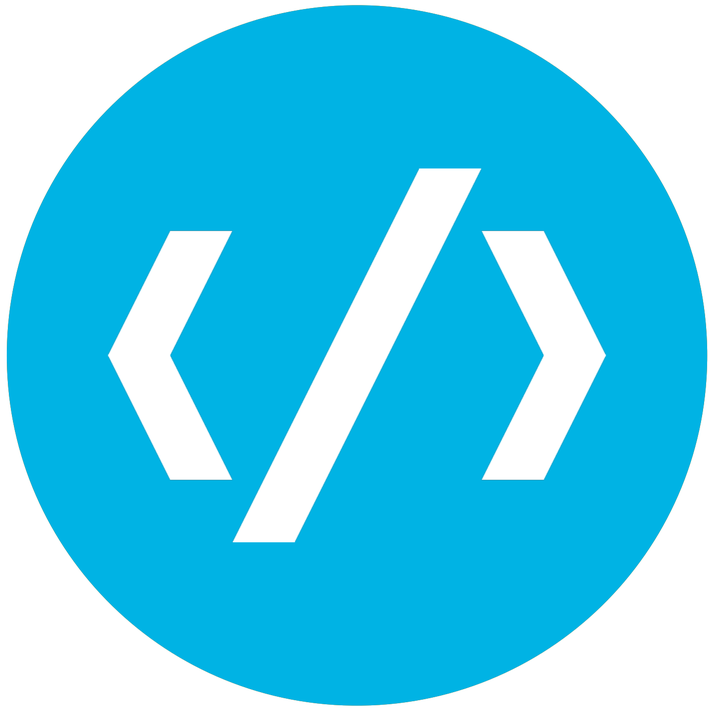

Dev Tools
History
Settings
Help
Console
1 GET /logstash-*/_search
2 {
3 "q"
4 }
5
query API
1


Dev Tools
History
Settings
Help
Console
1 GET /logstash-*/_search
2 {
3 "query": {
4 "q"
5 }
6 }
7
query_string API
simple_query_string API
1


Dev Tools
History
Settings
Help
Console
GET /logstash-*/_search
{
"query": {
"query_string": {
"default_field": "FIELD",
"query": "this AND that OR thus"
}
}
}

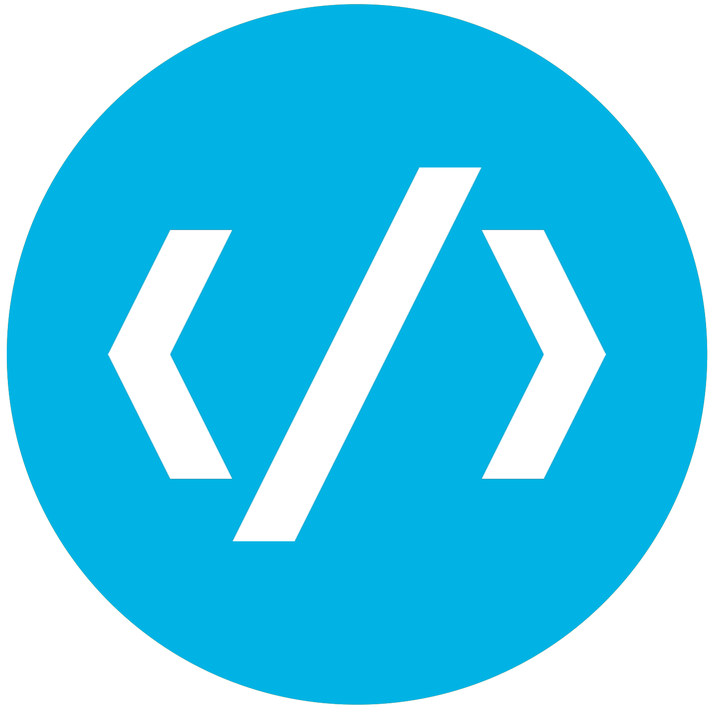

Dev Tools
History
Settings
Help
Console
GET /logstash-*/_search
{
"query": {
"query_string": {
"default_field": "_all",
"query": "apache OR nginx"
}
}
}


Dev Tools
History
Settings
Help
Console
GET /logstash-*/_search
{
"query": {
"query_string": {
"default_field": "_all",
"query": "apache OR nginx"
}
}
}
{
"took": 12,
"timed_out": false,
"_shards": {
"total": 100,
"successful": 100,
"failed": 0
},
"hits": {
"total": 28,
"max_score": 9.137492,
"hits": [
{
"_index": "logstash-prod-2016.05",
"_type": "log",
"_id": "AVjfkRSUyKCBlpIO2KdM",
"_score": 9.137492,
"_source": {
"@timestamp": 1463510588000,
"total_events": 264930,
"creation_time": 1343668126000,
"user_agent": {
"full_string": "TEST_INTERNET_AGENT",
"agent": "Other 0.0.0",
"agent_version": "0.0.0",
"os": {
"family": "Other",
"major": "0",



Mappings
Settings
Cluster
Indexing
Snapshots

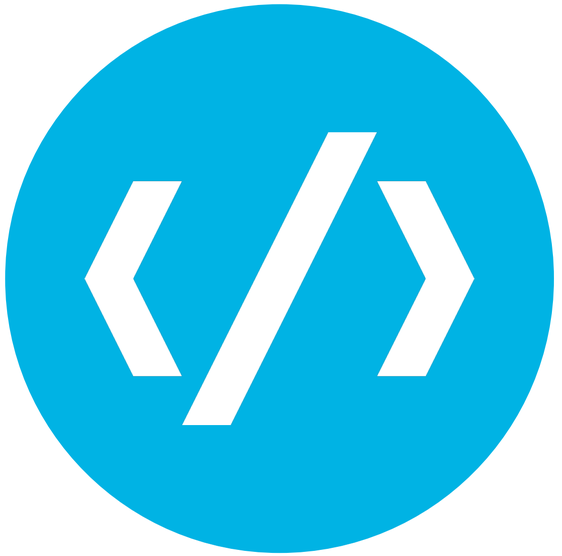
elastic



The Elasticsearch DSL

The entire thing. Made easy.

elastic



console

elastic

.what().if()

elastic

what if…

Elasticsearch was an amazing timeseries store?

elastic

what if…

Elasticsearch was an amazing timeseries store?

It is.

elastic

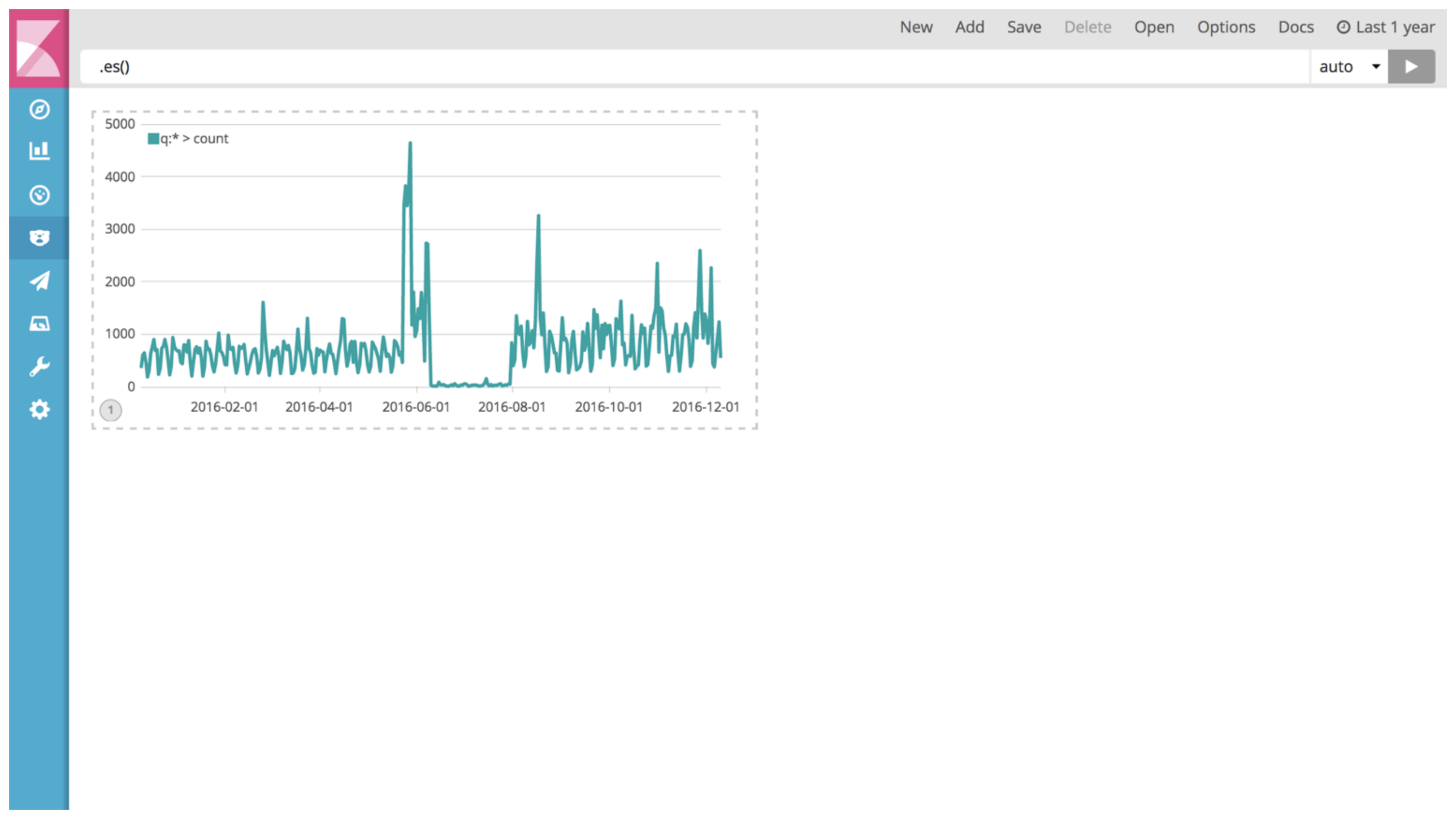
what if…

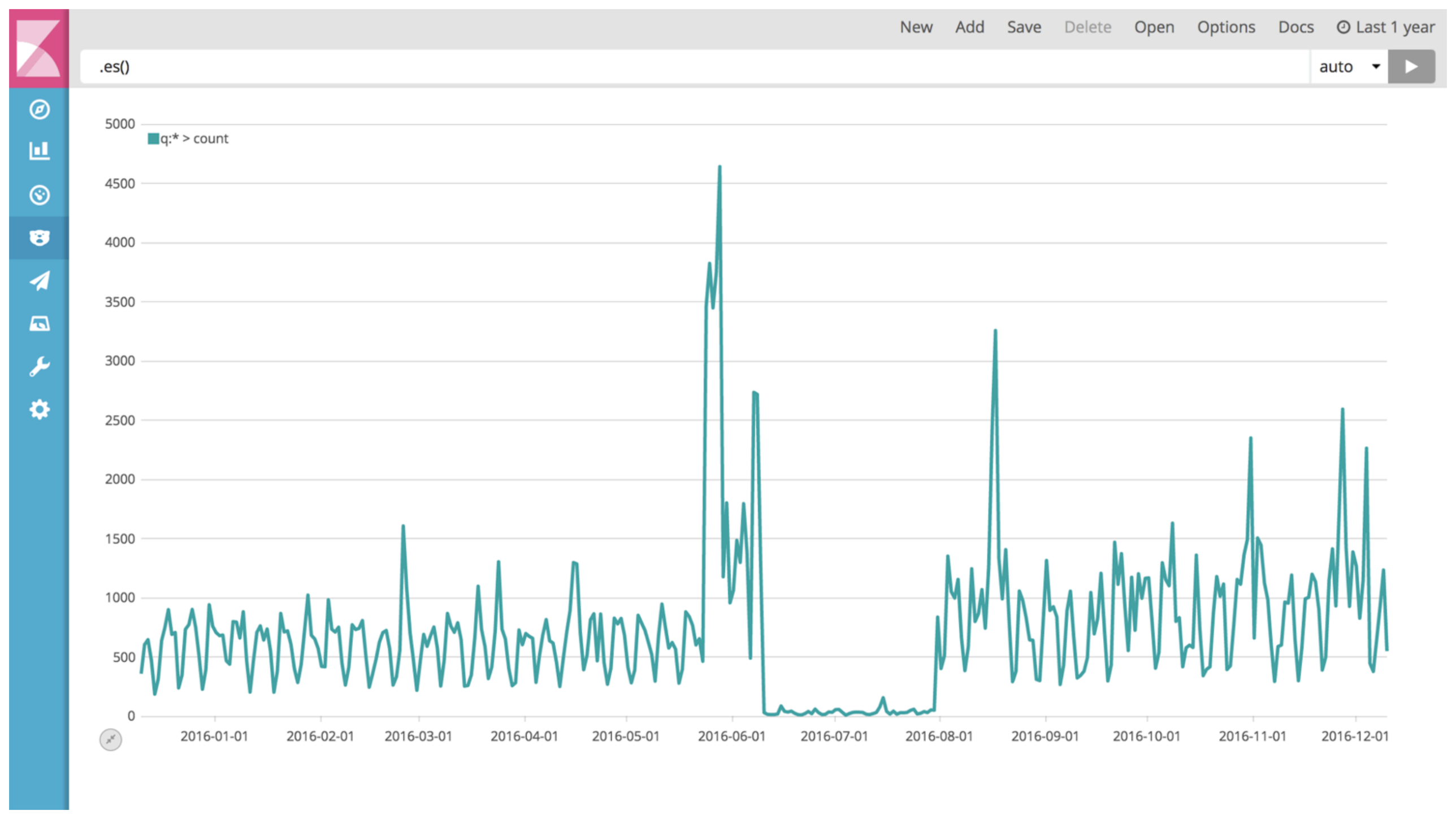
Elasticsearch was an amazing timeseries store?

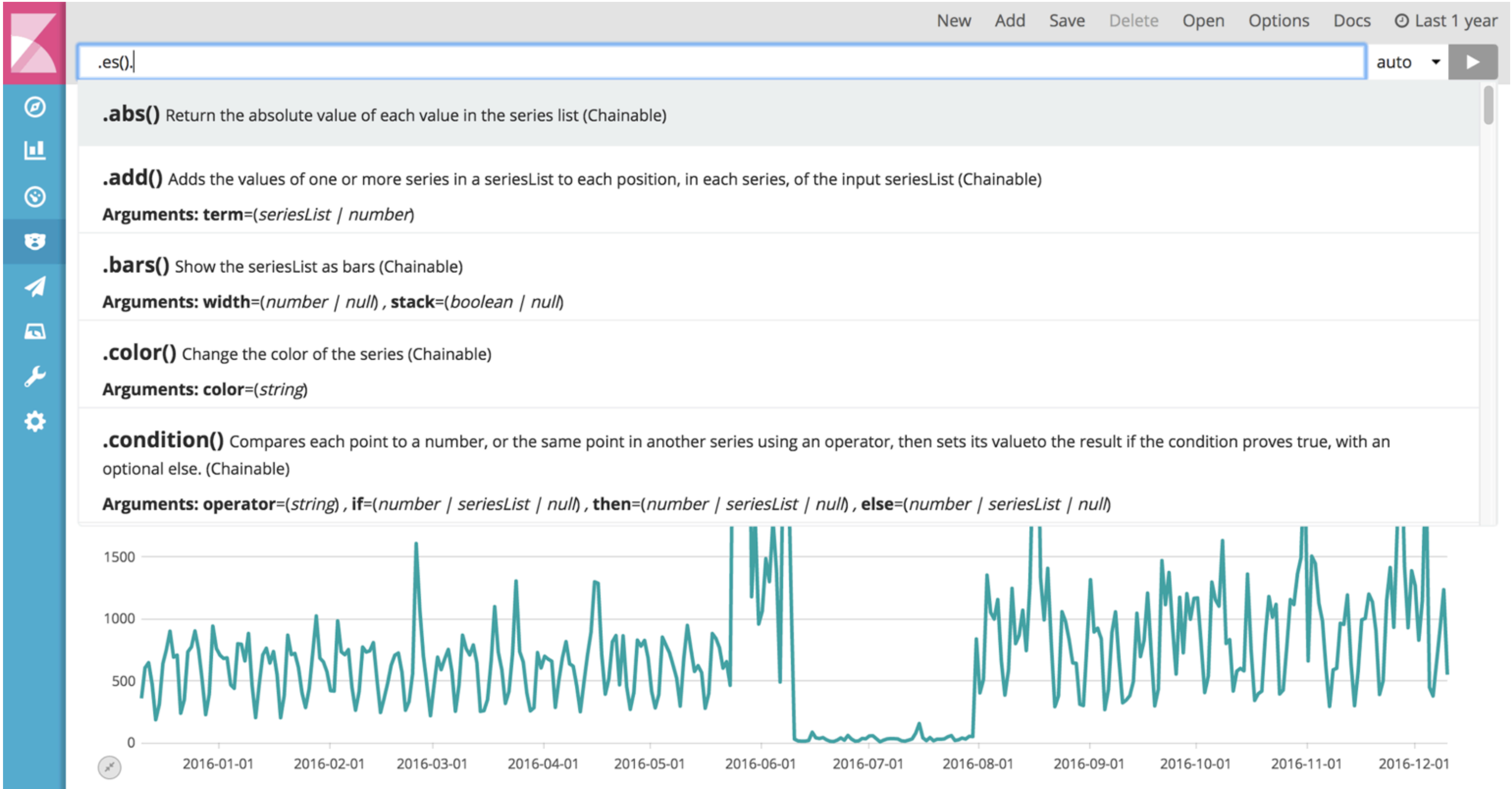
It is. And it has a UI.

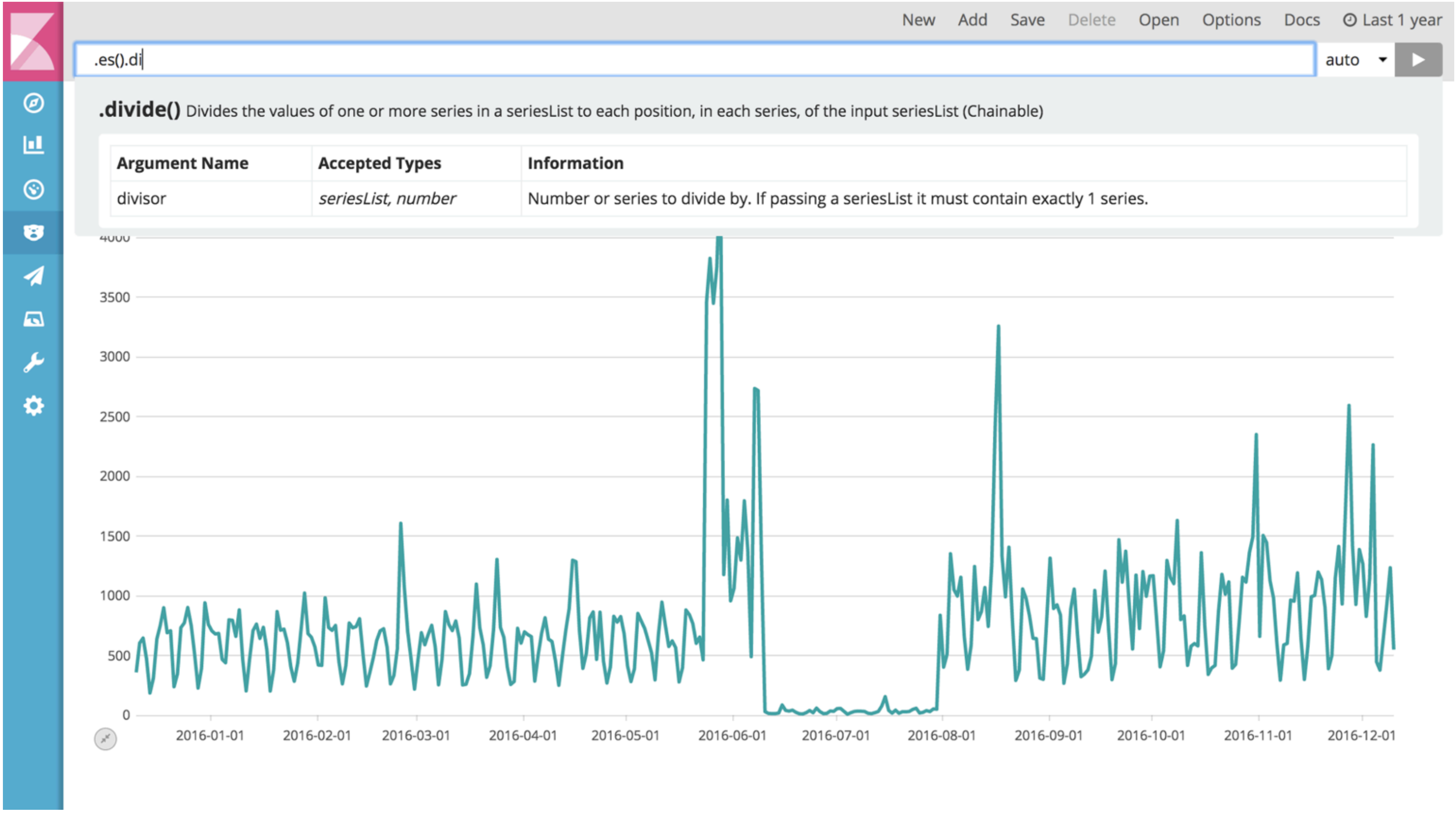
elastic

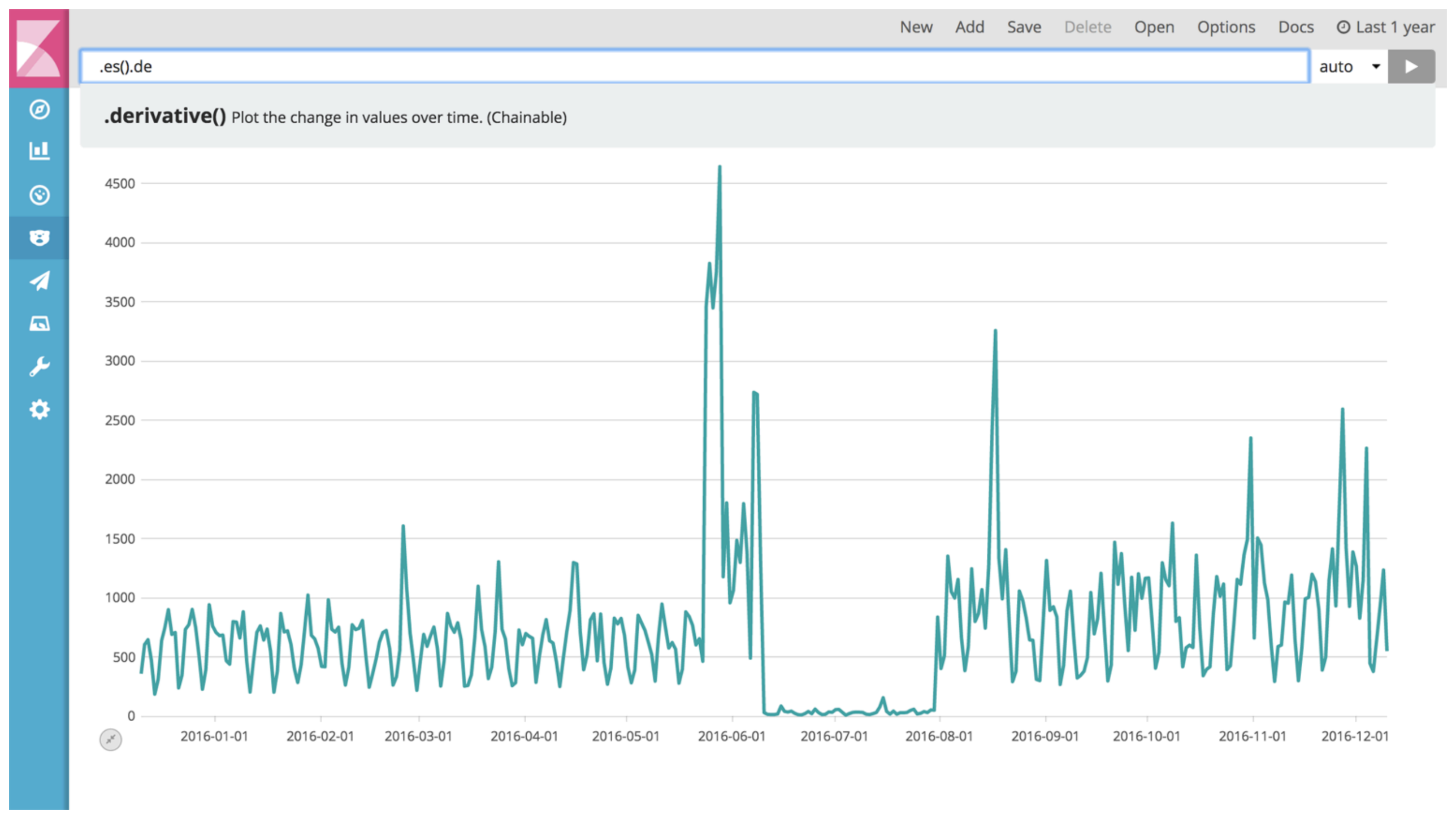
timelion


New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es()
auto
5000
4000
3000
2000
1000
0
q:* > count
2016-02-01
2016-04-01
2016-06-01
2016-08-01
2016-10-01
2016-12-01
1


New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es()
auto
q:* > count
5000
4500
4000
3500
3000
2500
2000
1500
1000
500
0
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01


New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es().
auto
.abs() Return the absolute value of each value in the series list (Chainable)
.add() Adds the values of one or more series in a seriesList to each position, in each series, of the input seriesList (Chainable)
Arguments: term=(seriesList | number)
.bars() Show the seriesList as bars (Chainable)
Arguments: width=(number | null) , stack=(boolean | null)
.color() Change the color of the series (Chainable)
Arguments: color=(string)
.condition() Compares each point to a number, or the same point in another series using an operator, then sets its valueto the result if the condition proves true, with an optional else. (Chainable)
Arguments: operator=(string) , if=(number | seriesList | null) , then=(number | seriesList | null) , else=(number | seriesList | null)
1500
1000
500
0
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

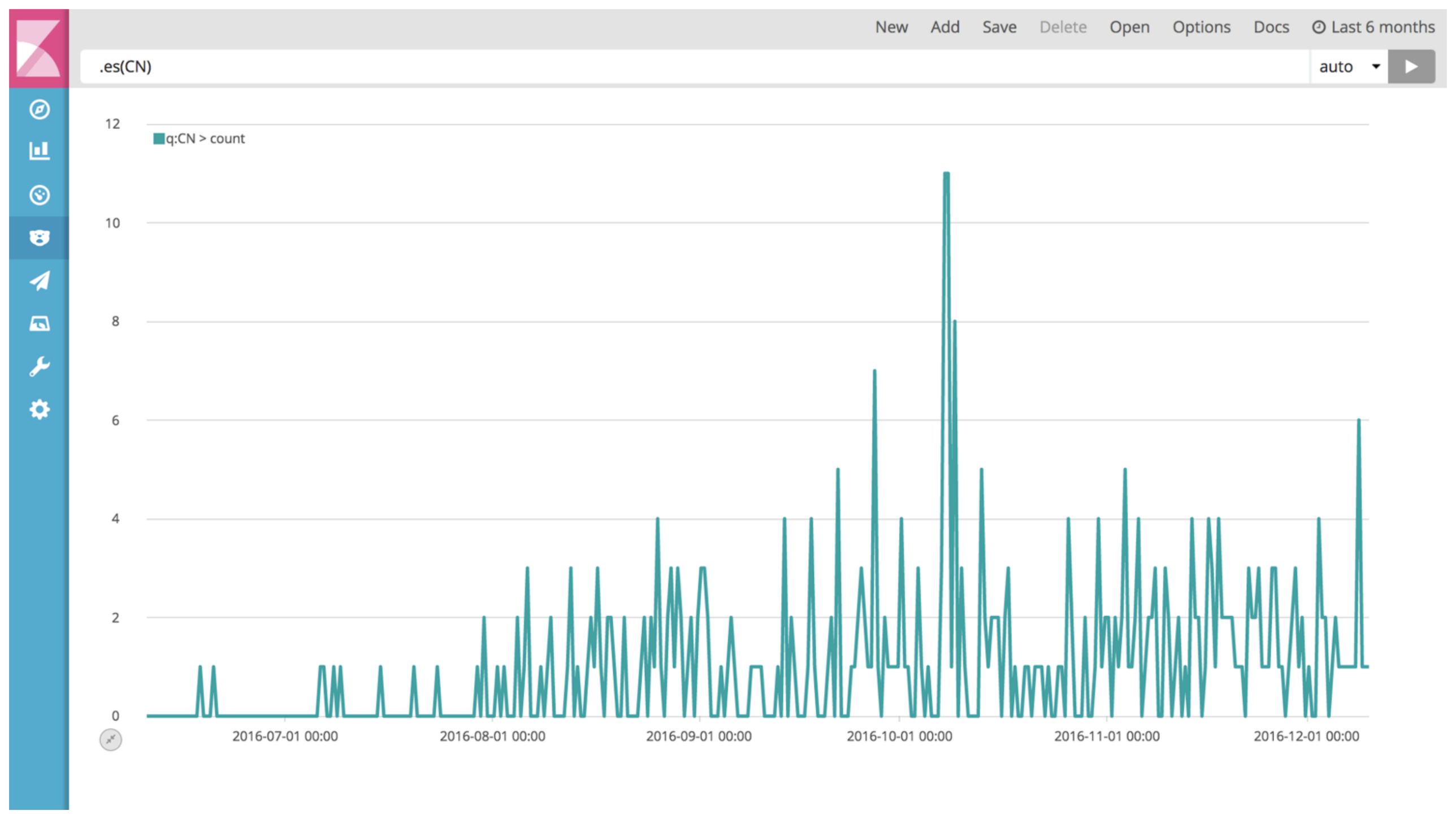

New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es().di
auto
.divide() Divides the values of one or more series in a seriesList to each position, in each series, of the input seriesList (Chainable)
Argument Name
Accepted Types
Information
divisor
seriesList, number
Number or series to divide by. If passing a seriesList it must contain exactly 1 series.
4000
3500
3000
2500
2000
1500
1000
500
0
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

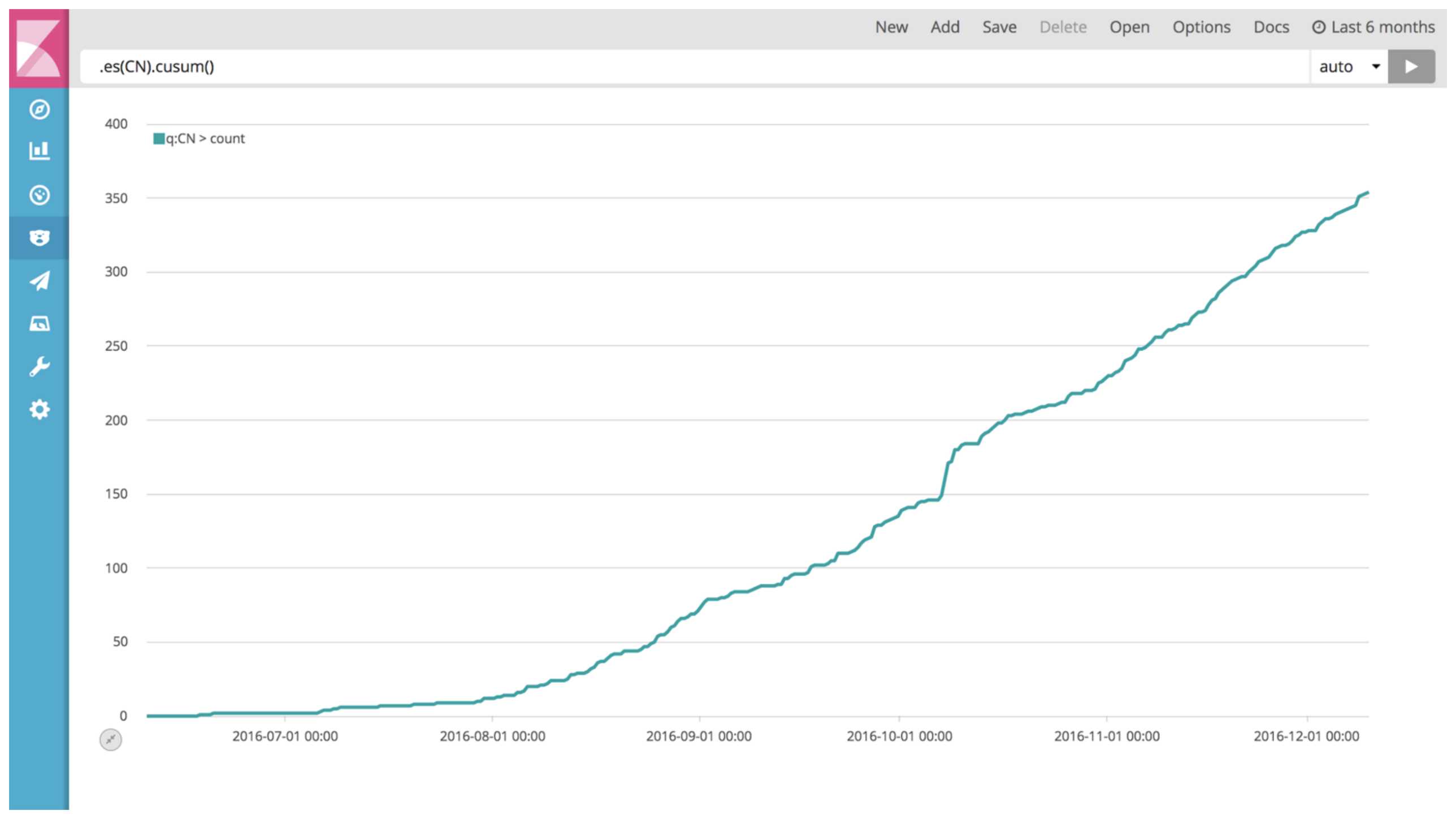

New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es().de
auto
.derivative() Plot the change in values over time. (Chainable)
4500
4000
3500
3000
2500
2000
1500
1000
500
0
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

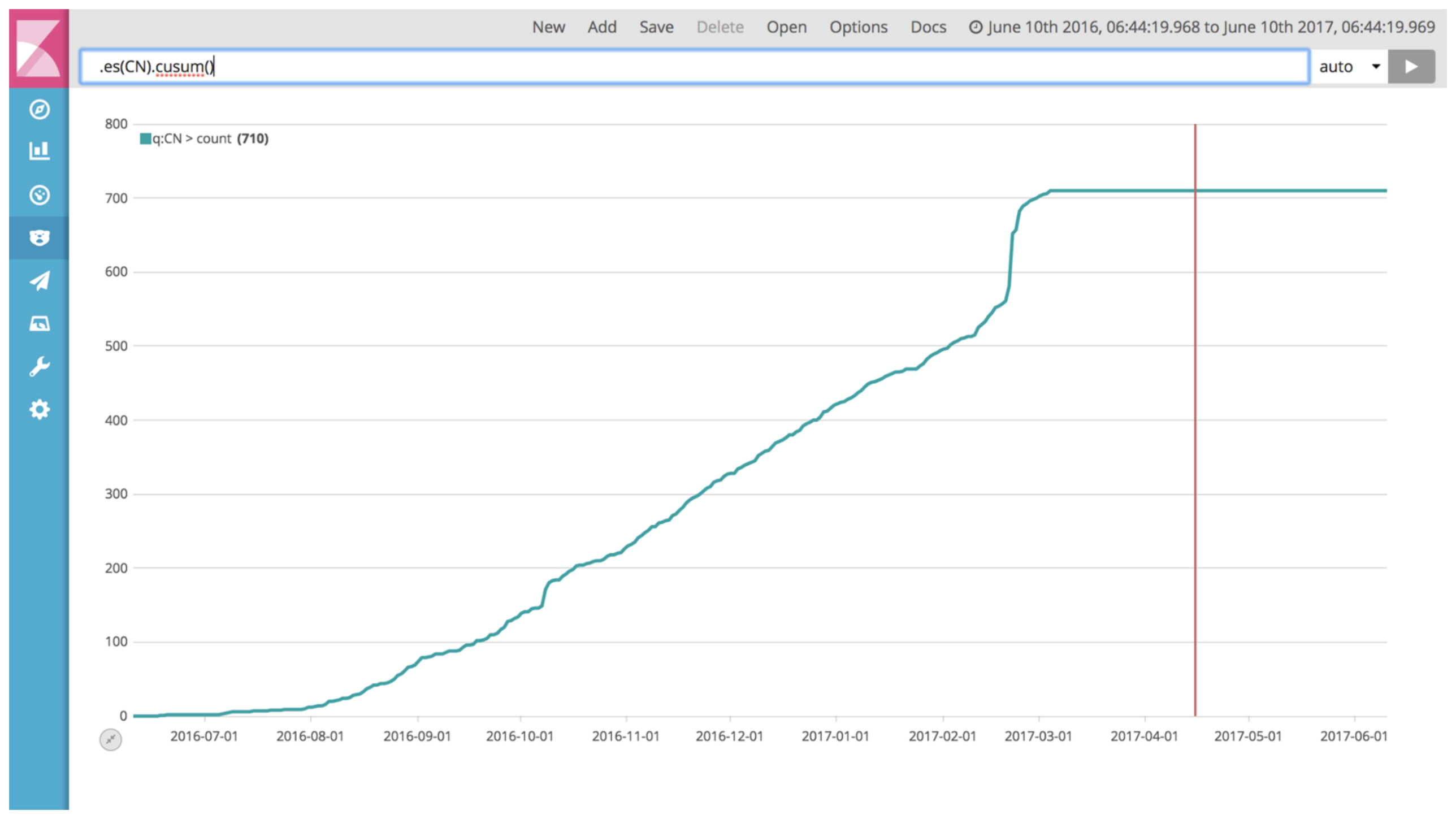

New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es().derivative()
auto
q:* > count
4000
3000
2000
1000
0
-1000
-2000
-3000
-4000
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

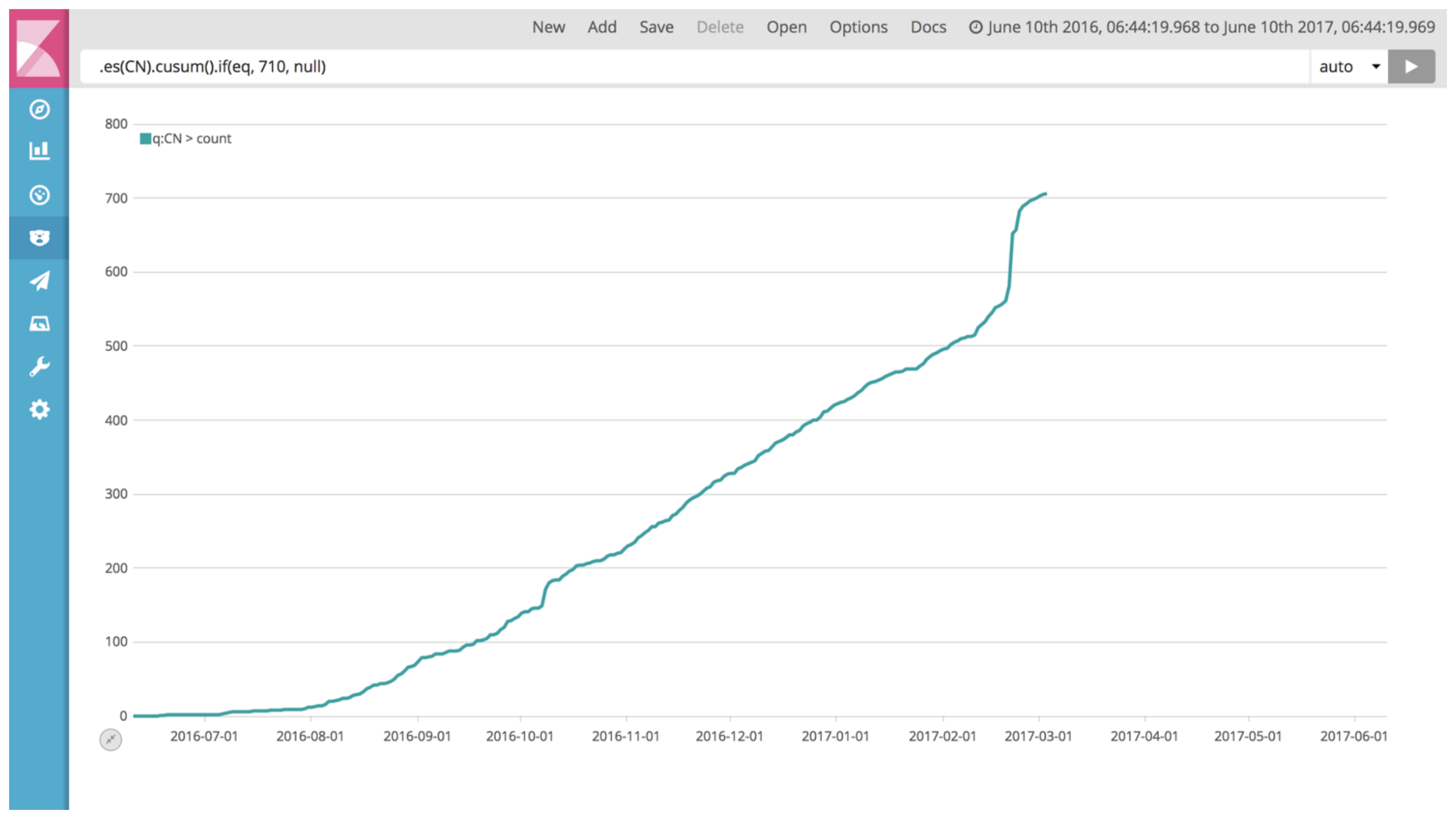

New
Add
Save
Delete
Open
Options
Docs
Last 1 year
(.es(CN), .es(TW)).derivative()
auto
q:CN > count
q:TW > count
40
30
20
10
0
-10
-20
-30
-40
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01


New
Add
Save
Delete
Open
Options
Docs
Last 1 year
(.es(CN), .es(TW), .es(HK)).derivative()
auto
q:CN > count
q:TW > count
q:HK > count
300
200
100
0
-100
-200
-300
-400
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01


New
Add
Save
Delete
Open
Options
Docs
Last 6 months
.es(CN)
auto
q:CN > count
12
10
8
6
4
2
0
2016-07-01 00:00
2016-08-01 00:00
2016-09-01 00:00
2016-10-01 00:00
2016-11-01 00:00
2016-12-01 00:00


New
Add
Save
Delete
Open
Options
Docs
Last 6 months
.es(CN).cusum()
auto
400
350
300
250
200
150
100
50
0
q:CN > count
2016-07-01 00:00
2016-08-01 00:00
2016-09-01 00:00
2016-10-01 00:00
2016-11-01 00:00
2016-12-01 00:00


New
Add
Save
Delete
Open
Options
Docs
June 10th 2016, 06:44:19.968 to June 10th 2017, 06:44:19.969
.es(CN).cusum()
auto
q:CN > count (710)
800
700
600
500
400
300
200
100
0
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01
2017-01-01
2017-02-01
2017-03-01
2017-04-01
2017-05-01
2017-06-01

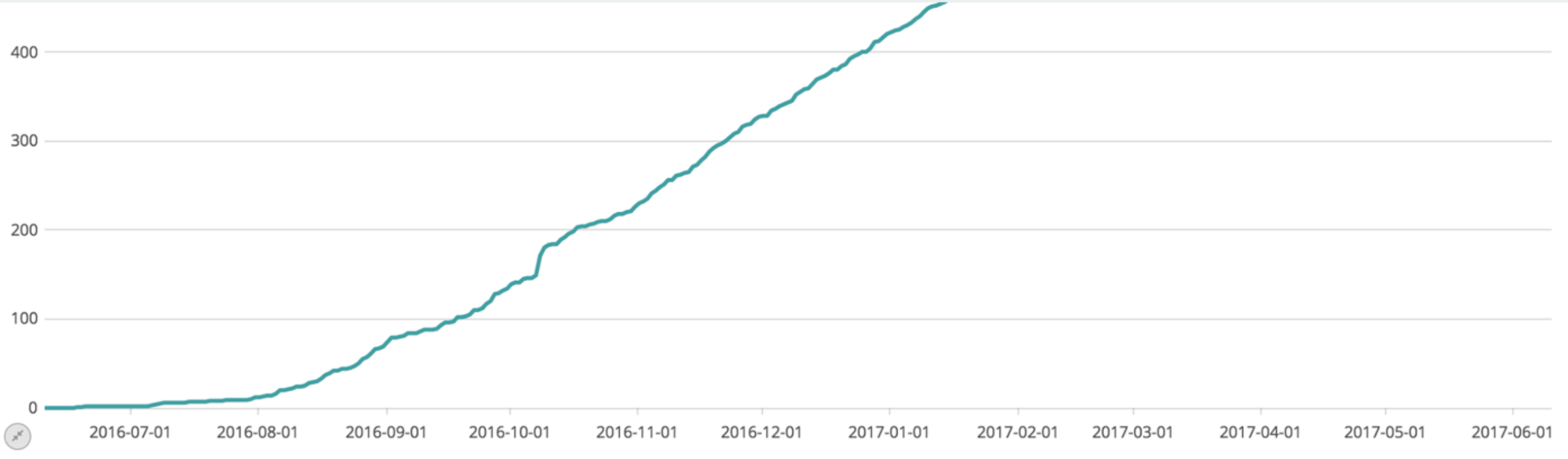

New
Add
Save
Delete
Open
Options
Docs
June 10th 2016, 06:44:19.968 to June 10th 2017, 06:44:19.969
.es(CN).cusum().if(eq, 710, null)
auto
800
700
600
500
400
300
200
100
0
q:CN > count
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01
2017-01-01
2017-02-01
2017-03-01
2017-04-01
2017-05-01
2017-06-01

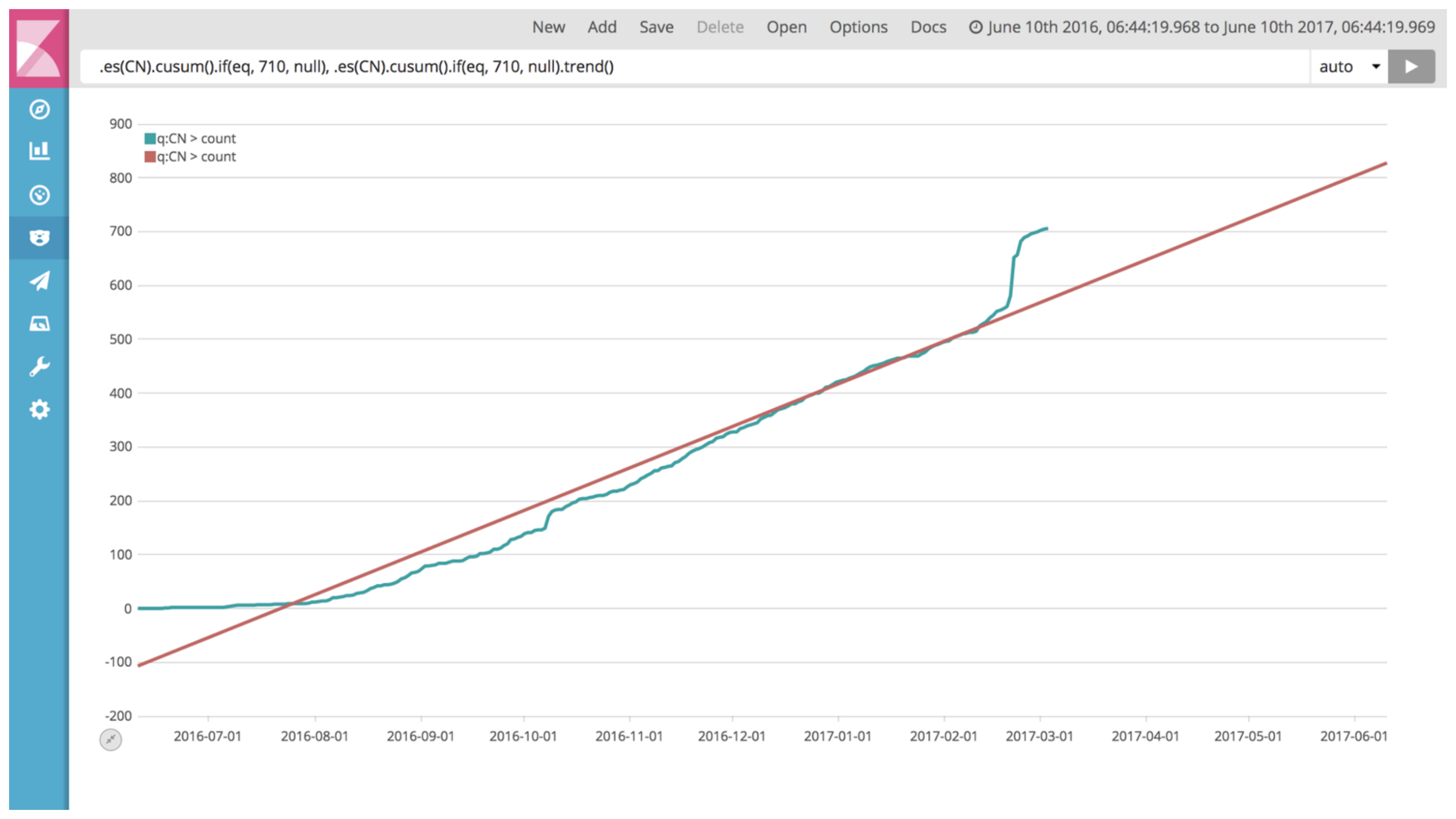
.es(CN).cusum().if(eq, 710, null), .es(CN).cusum().if(eq, 710, null).tre

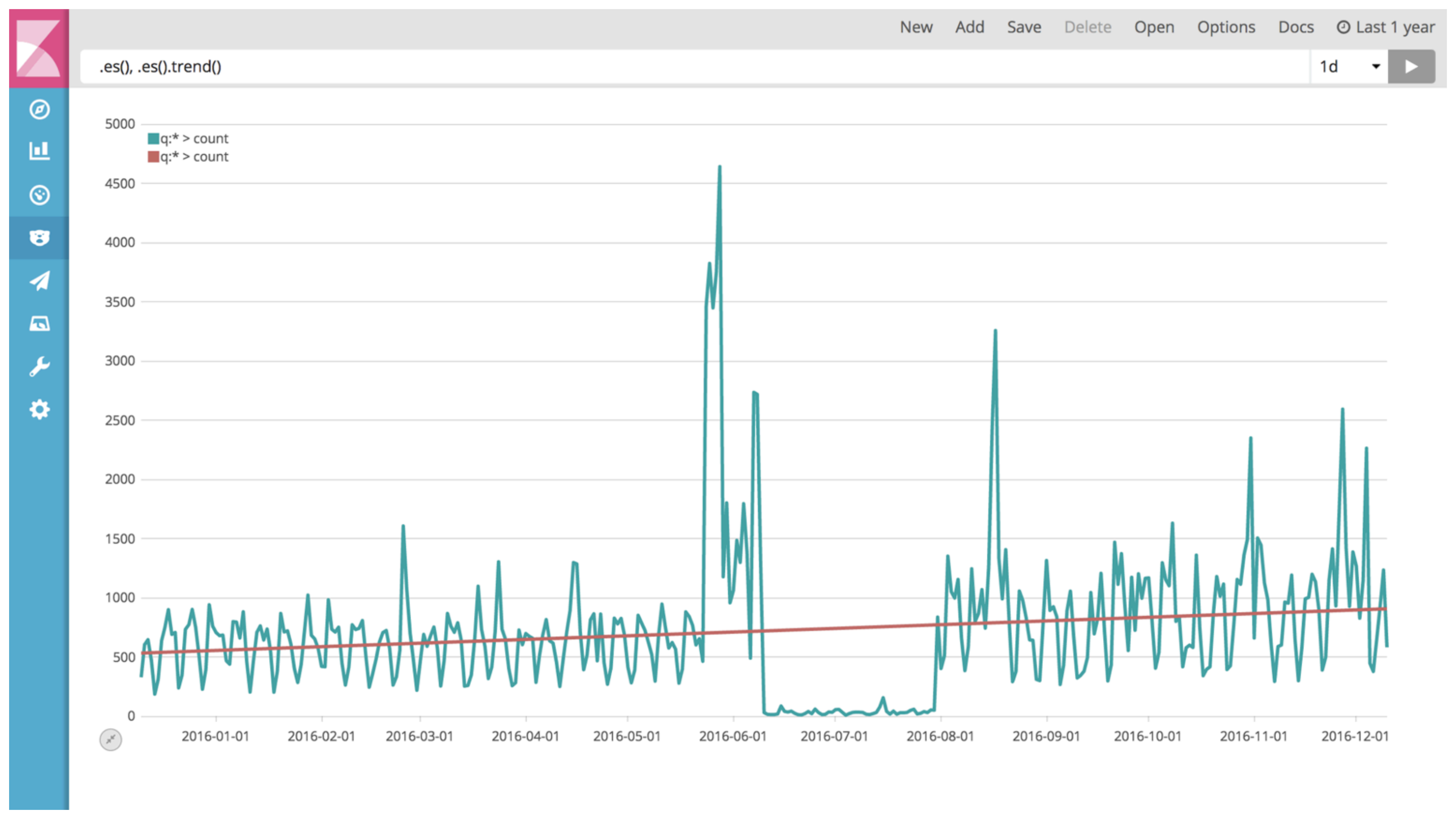
auto

.trend() Draws a trend line using a specified regression algorithm (Chainable)

Argument Name	Accepted Types	Information
mode	*string*	The algorithm to use for generating the trend line. One of: linear, log
start	*number, null*	Where to start calculating from the beginning or end. For example -10 would start calculating 10 points from the end, +15 would start 15 points from the beginning. Default: 0
end	*number, null*	Where to stop calculating from the beginning or end. For example -10 would stop calculating 10 points from the end, +15 would stop 15 points from the beginning. Default: 0



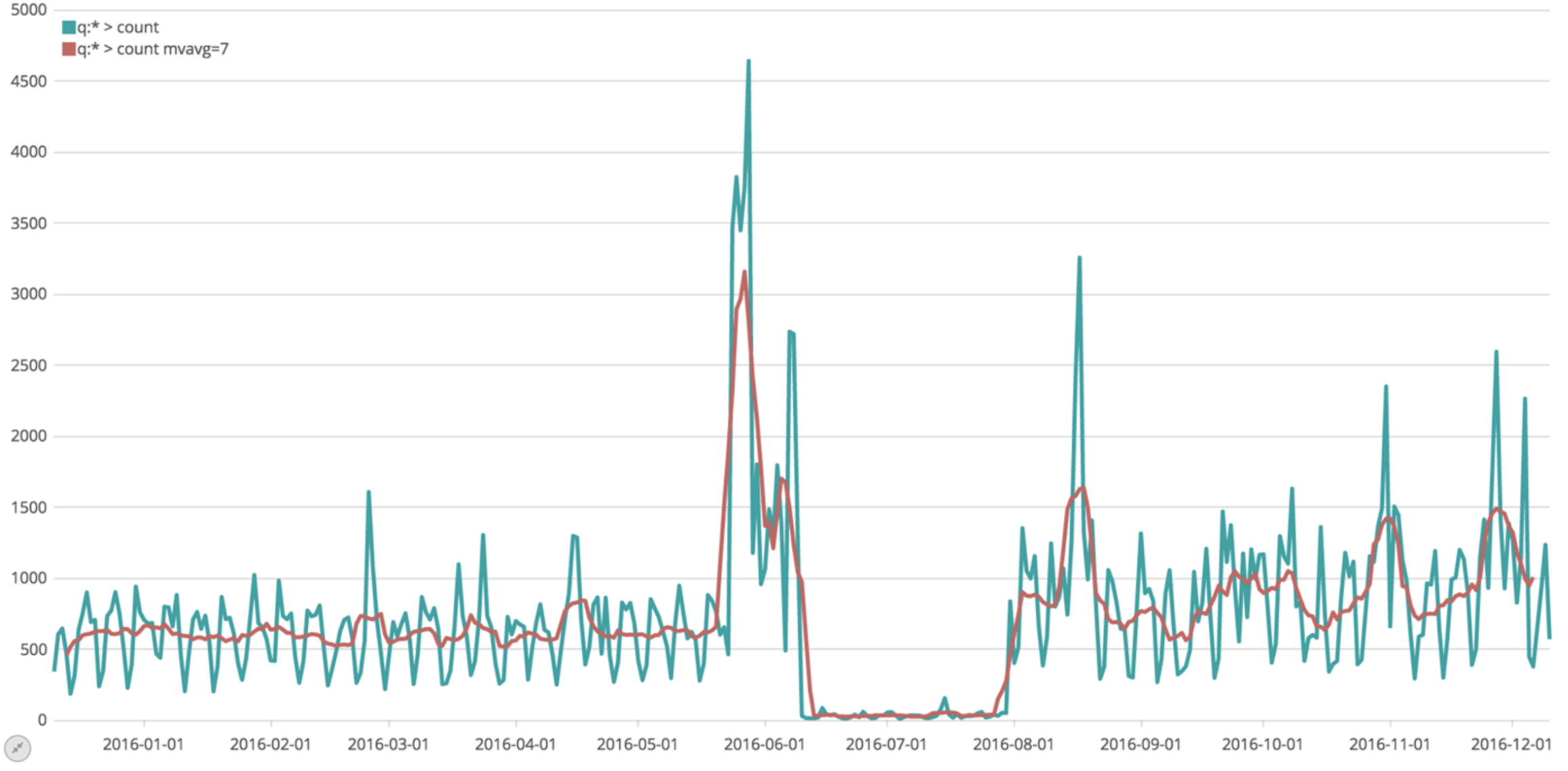

New
Add
Save
Delete
Open
Options
Docs
June 10th 2016, 06:44:19.968 to June 10th 2017, 06:44:19.969
.es(CN).cusum().if(eq, 710, null), .es(CN).cusum().if(eq, 710, null).trend()
auto
q:CN > count
q:CN > count
900
800
700
600
500
400
300
200
100
0
-100
-200
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01
2017-01-01
2017-02-01
2017-03-01
2017-04-01
2017-05-01
2017-06-01

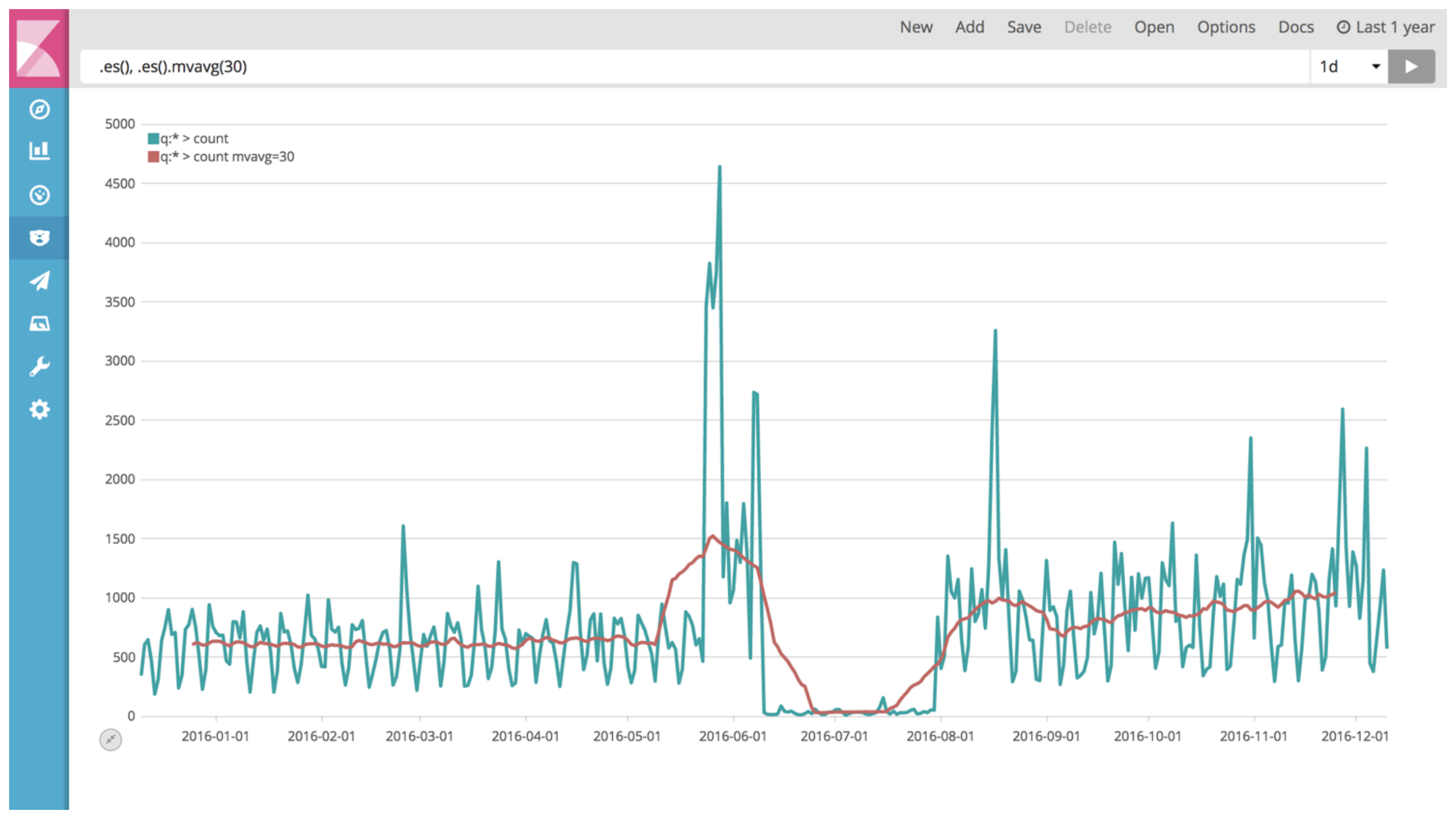

New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(), .es().trend()
1d
q:* > count
q:* > count
5000
4500
4000
3500
3000
2500
2000
1500
1000
500
0
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

New Add Save Delete Open Options Docs Last 1 year

.es(), .es().mvavg(7)

1d

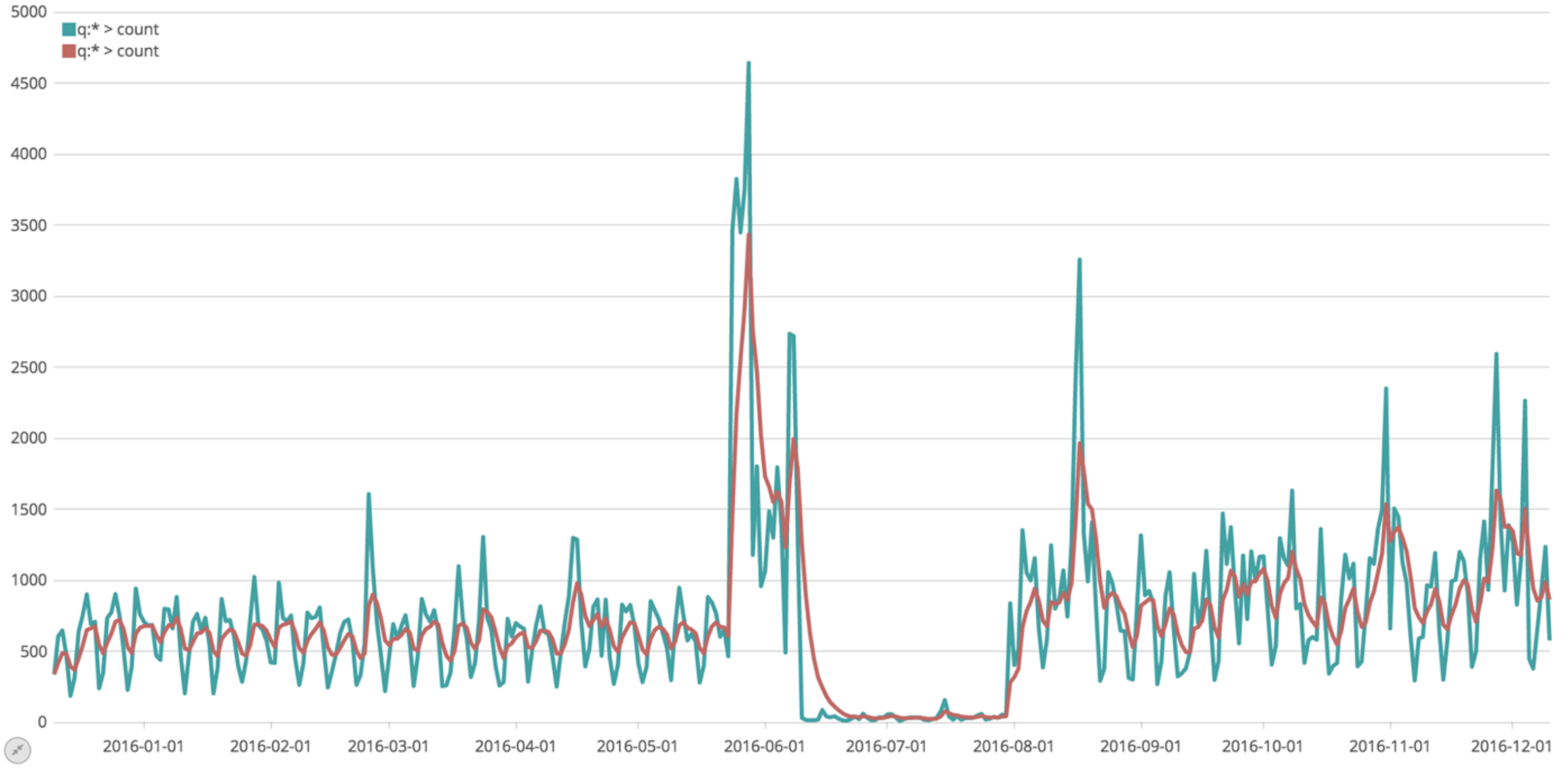


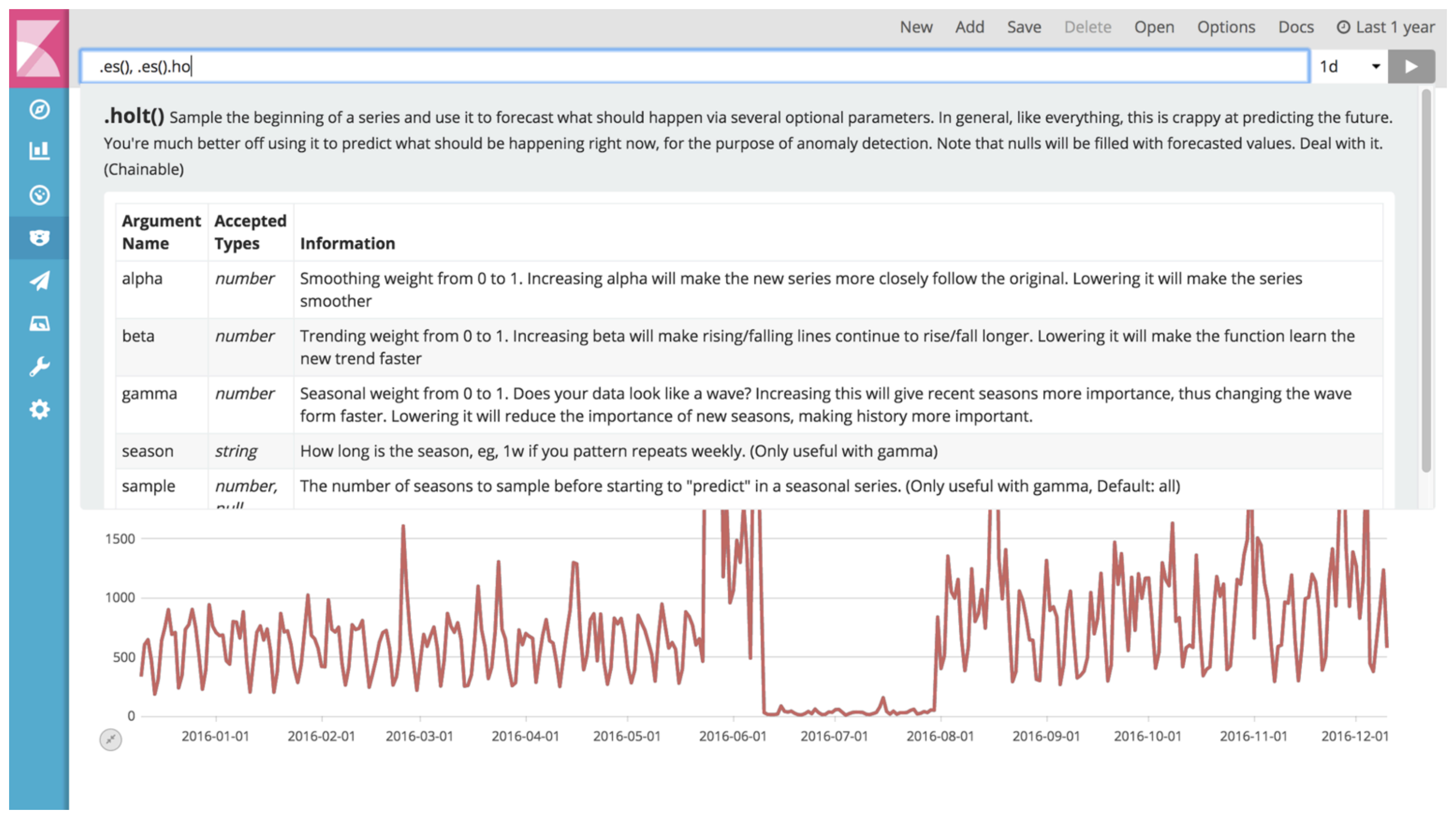

New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(), .es().mvavg(30)
1d
q:* > count
q:* > count mvavg=30
5000
4500
4000
3500
3000
2500
2000
1500
1000
500
0
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

New Add Save Delete Open Options Docs Last 1 year

.es(), .es().holt(0.3)

1d

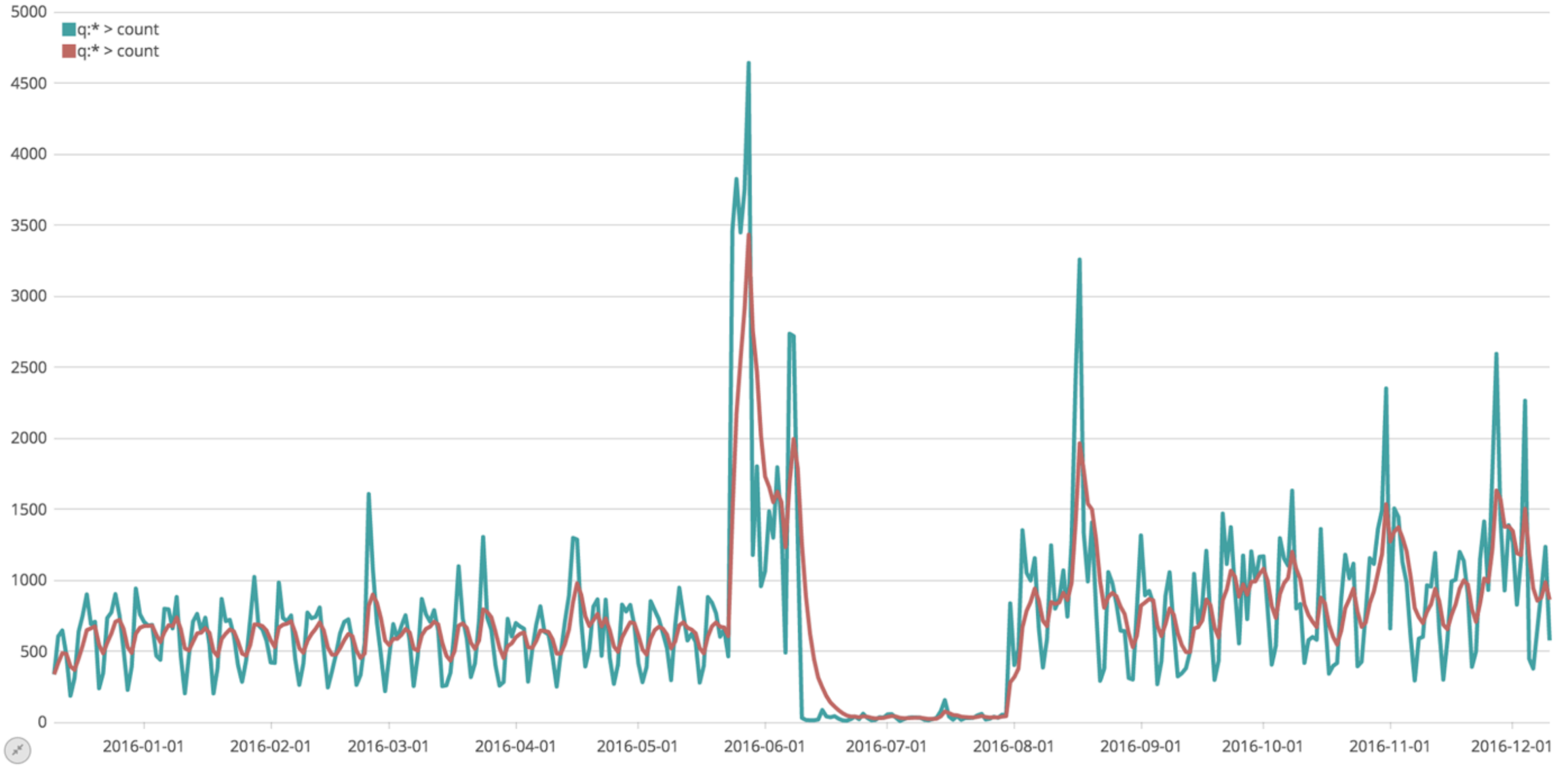


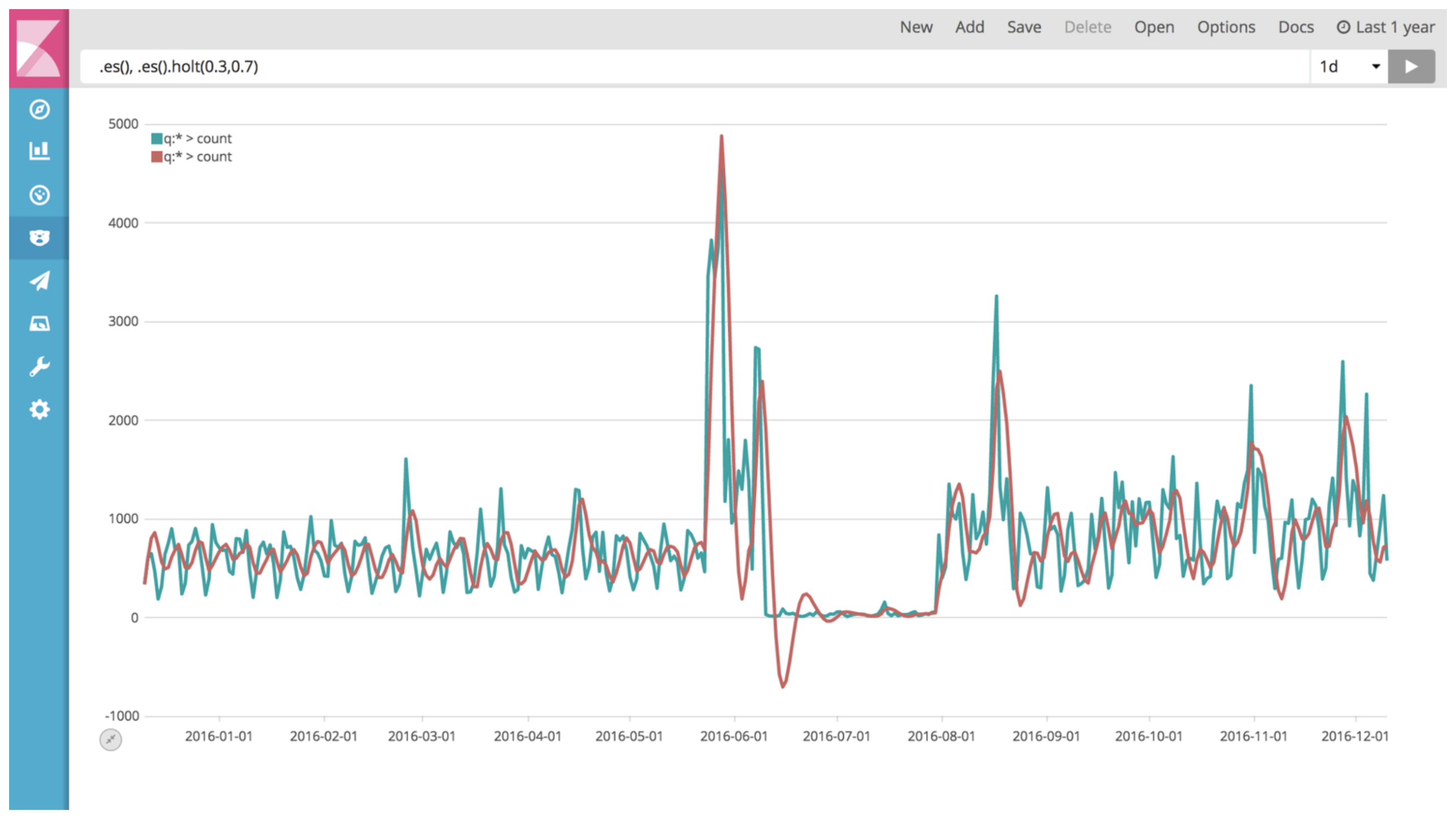

New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(), .es().ho
1d
.holt() Sample the beginning of a series and use it to forecast what should happen via several optional parameters. In general, like everything, this is crappy at predicting the future. You're much better off using it to predict what should be happening right now, for the purpose of anomaly detection. Note that nulls will be filled with forecasted values. Deal with it. (Chainable)
Argument Name
Accepted Types
Information
alpha
number
Smoothing weight from 0 to 1. Increasing alpha will make the new series more closely follow the original. Lowering it will make the series smoother
beta
number
Trending weight from 0 to 1. Increasing beta will make rising/falling lines continue to rise/fall longer. Lowering it will make the function learn the new trend faster
gamma
number
Seasonal weight from 0 to 1. Does your data look like a wave? Increasing this will give recent seasons more importance, thus changing the wave form faster. Lowering it will reduce the importance of new seasons, making history more important.
season
string
How long is the season, eg, 1w if you pattern repeats weekly. (Only useful with gamma)
sample
number,
The number of seasons to sample before starting to "predict" in a seasonal series. (Only useful with gamma, Default: all)
1500
1000
500
0
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

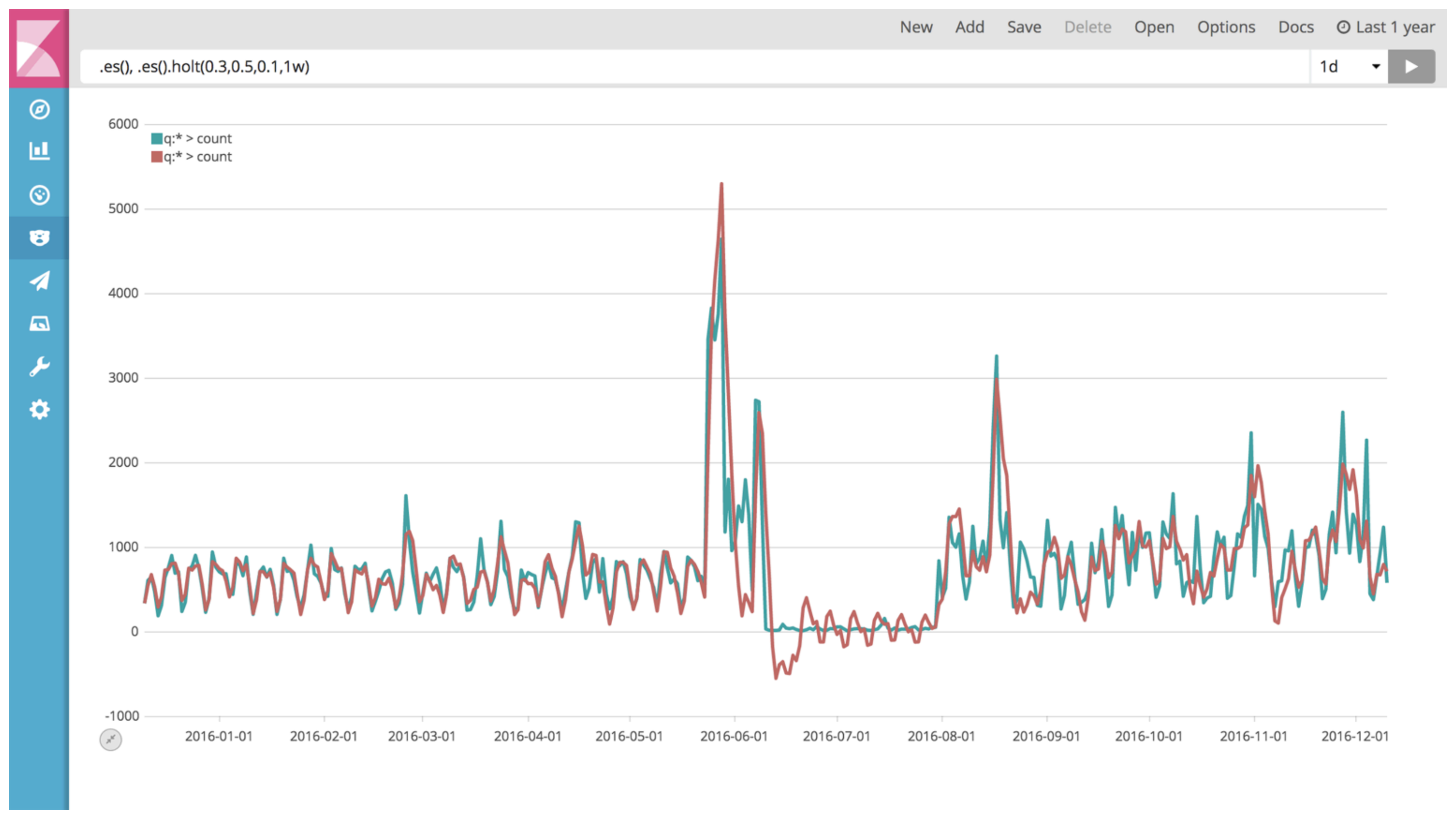
New Add Save Delete Open Options Docs Last 1 year

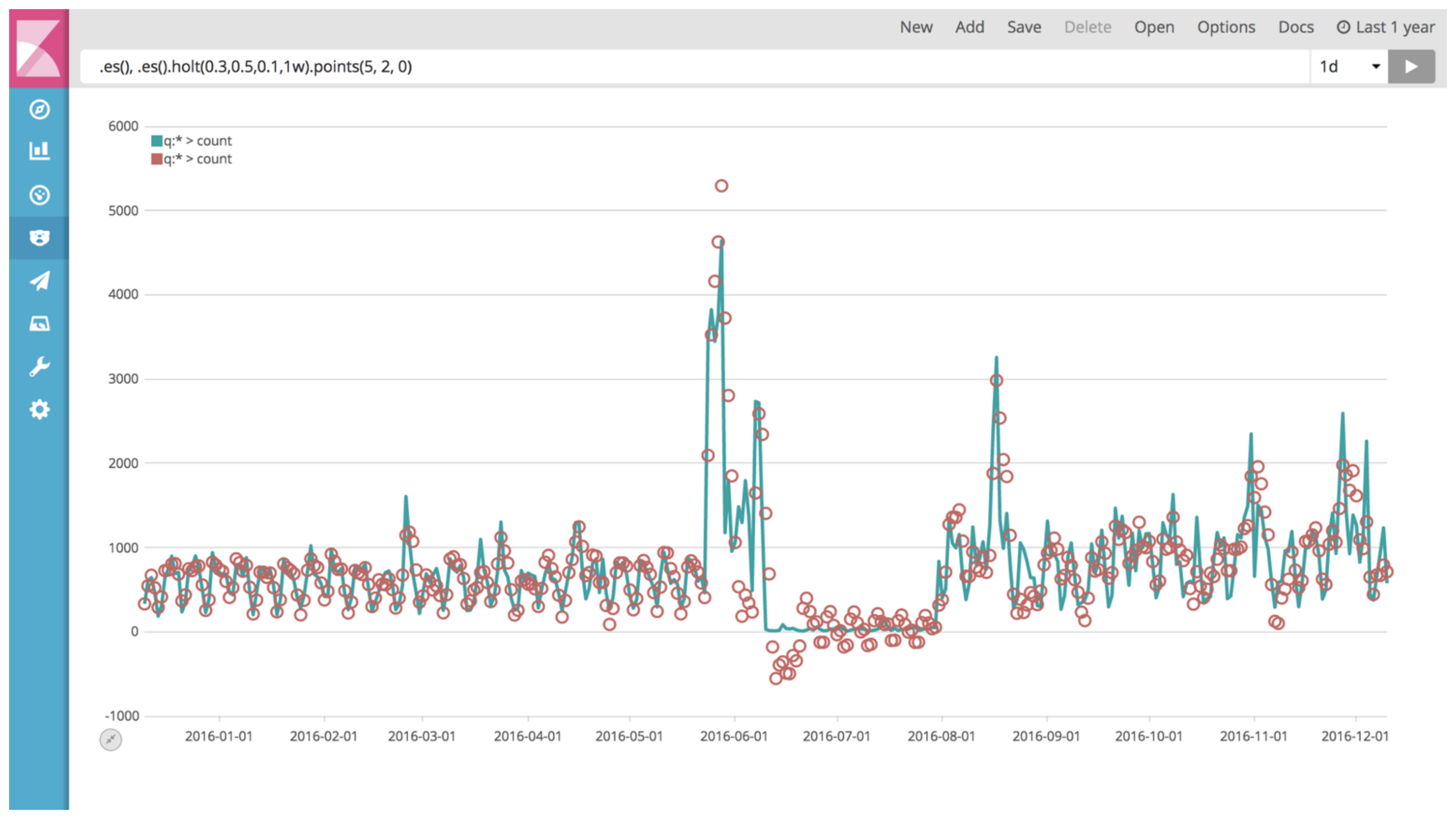
.es(), .es().holt(0.3)

1d




New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(), .es().holt(0.3,0.7)
1d
q:* > count
q:* > count
5000
4000
3000
2000
1000
0
-1000
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01


New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(), .es().holt(0.3,0.5,0.1,1w)
1d
q:* > count
q:* > count
6000
5000
4000
3000
2000
1000
0
-1000
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01


New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(), .es().holt(0.3,0.5,0.1,1w).points(5, 2, 0)
1d
q:* > count
q:* > count
6000
5000
4000
3000
2000
1000
0
-1000
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

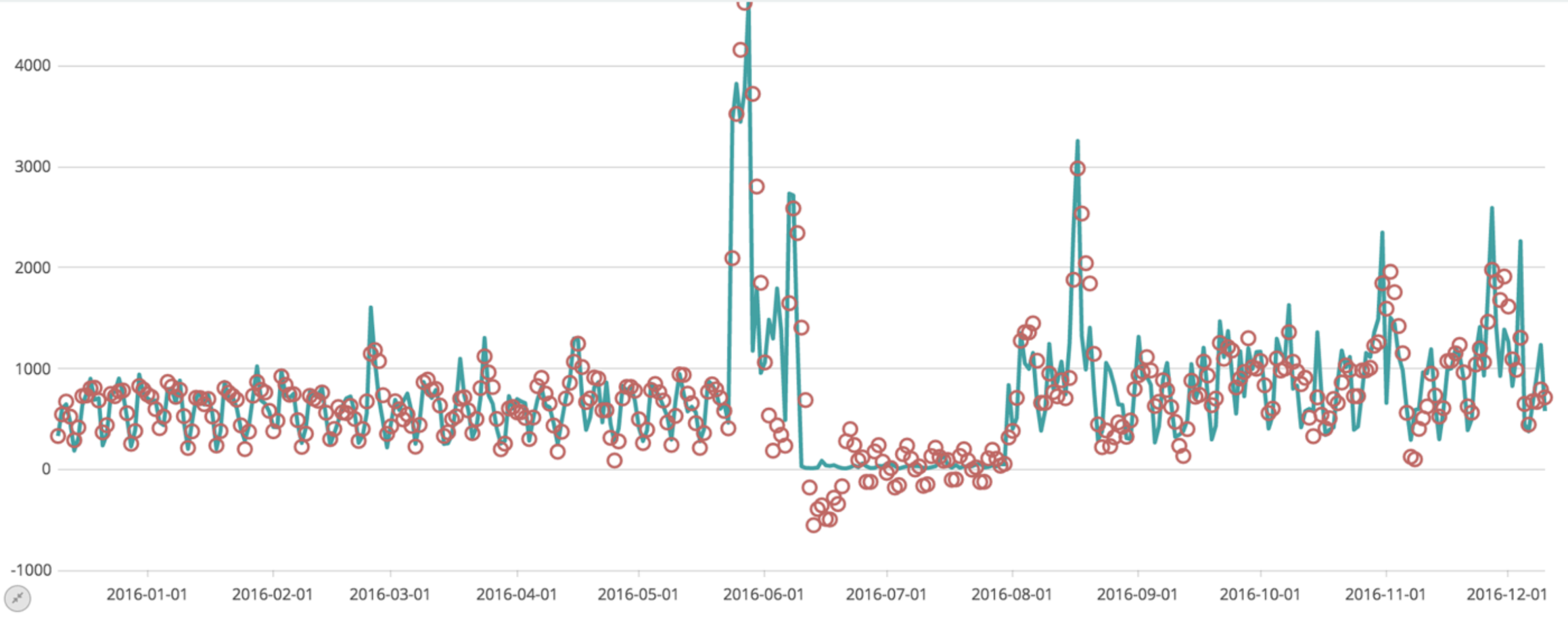
New Add Save Delete Open Options Docs Last 1 year

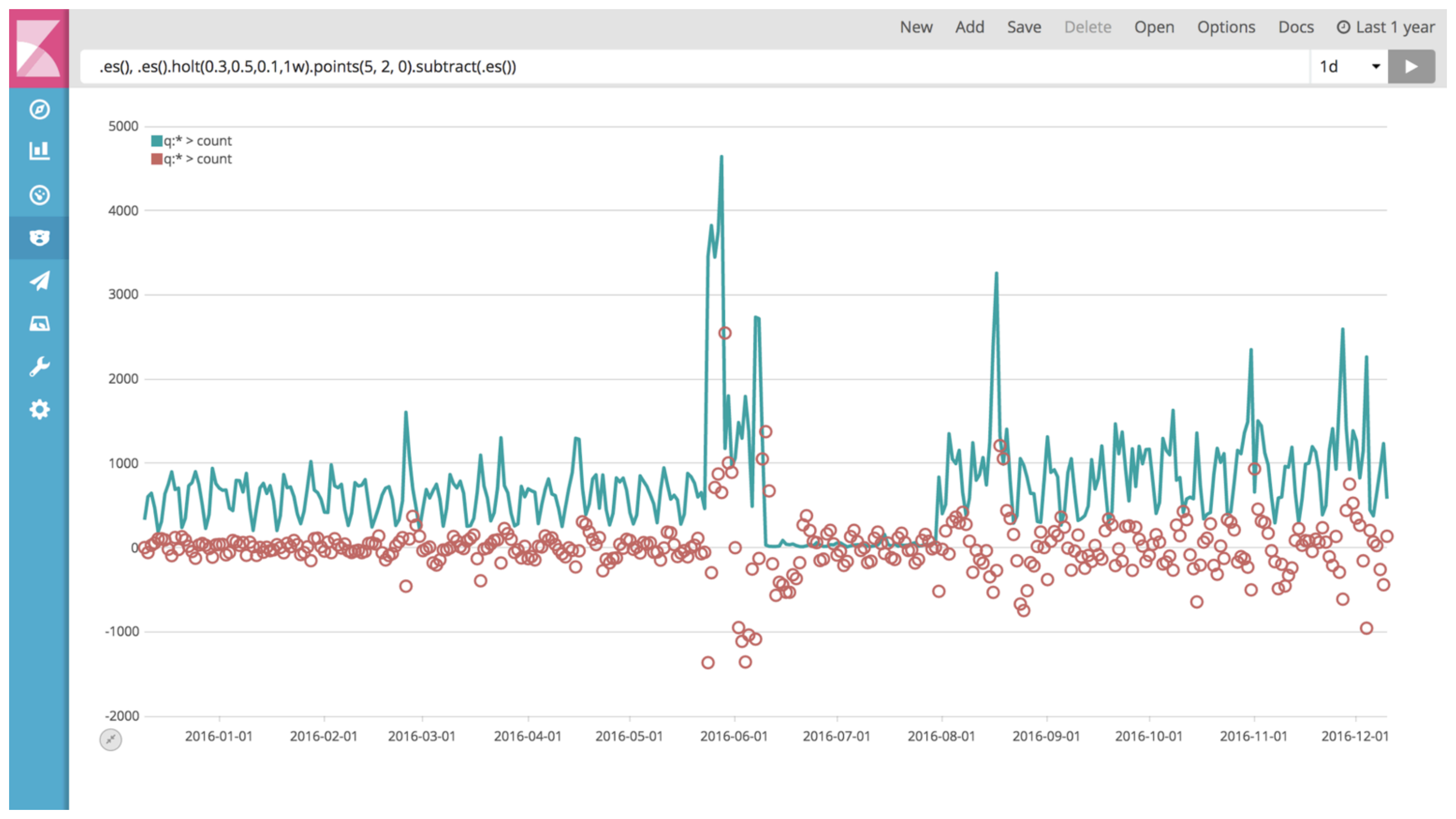
.es(), .es().holt(0.3,0.5,0.1,1w).points(5, 2, 0).sub

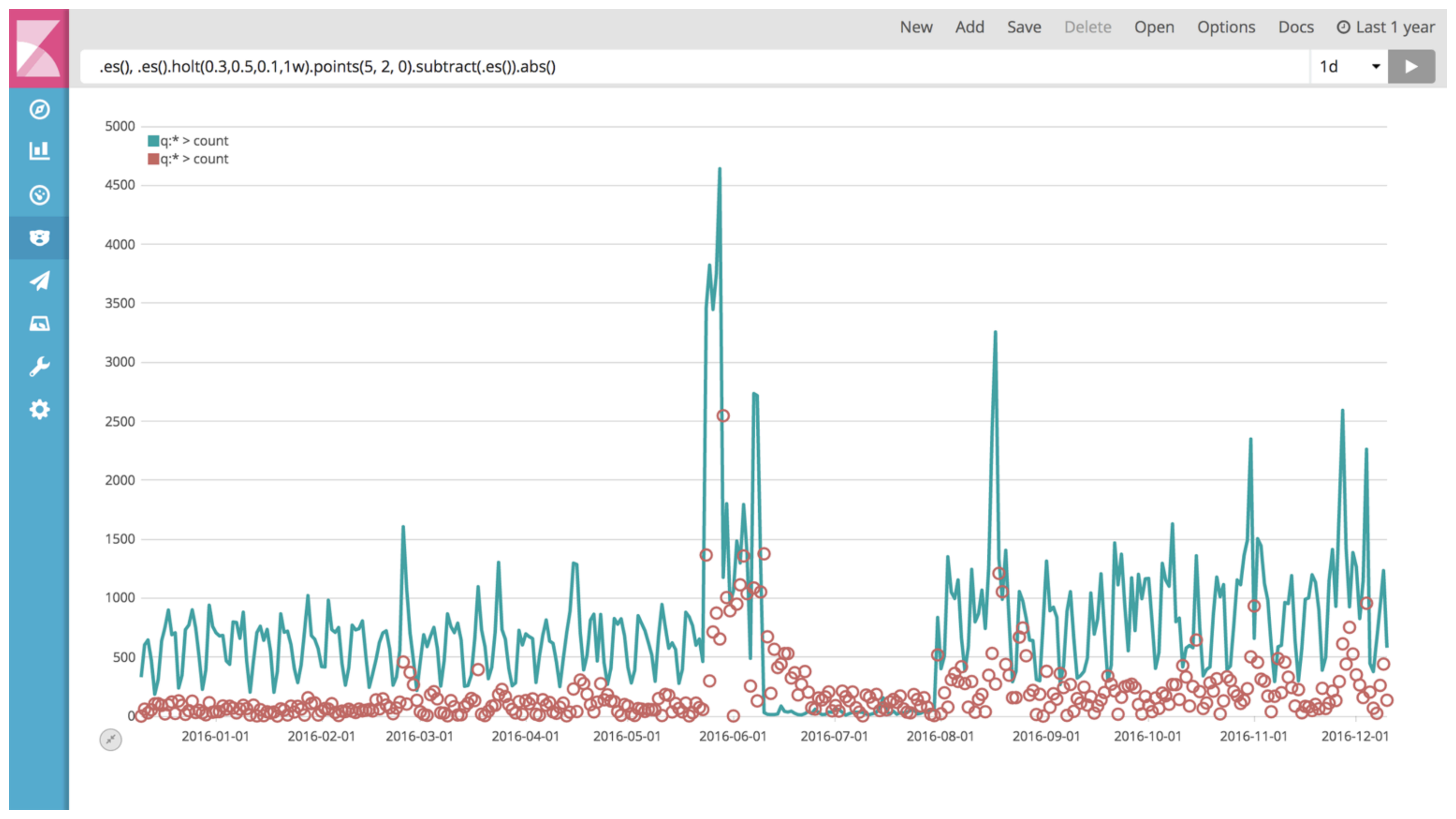
1d

.subtract() Subtract the values of one or more series in a seriesList to each position, in each series, of the input seriesList (Chainable)

Argument Name	Accepted Types	Information
term	*seriesList, number*	Number or series to subtract from input. If passing a seriesList it must contain exactly 1 series.




New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(), .es().holt(0.3,0.5,0.1,1w).points(5, 2, 0).subtract(.es())
1d
q:* > count
q:* > count
5000
4000
3000
2000
1000
0
-1000
-2000
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01


New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(), .es().holt(0.3,0.5,0.1,1w).points(5, 2, 0).subtract(.es()).abs()
1d
q:* > count
q:* > count
5000
4500
4000
3500
3000
2500
2000
1500
1000
500
0
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

.**if**(lt, 1000, null, .es())

elastic

.if(**lt**, 1000, null, .es())

elastic

.if(lt, **1000**, null, .es())

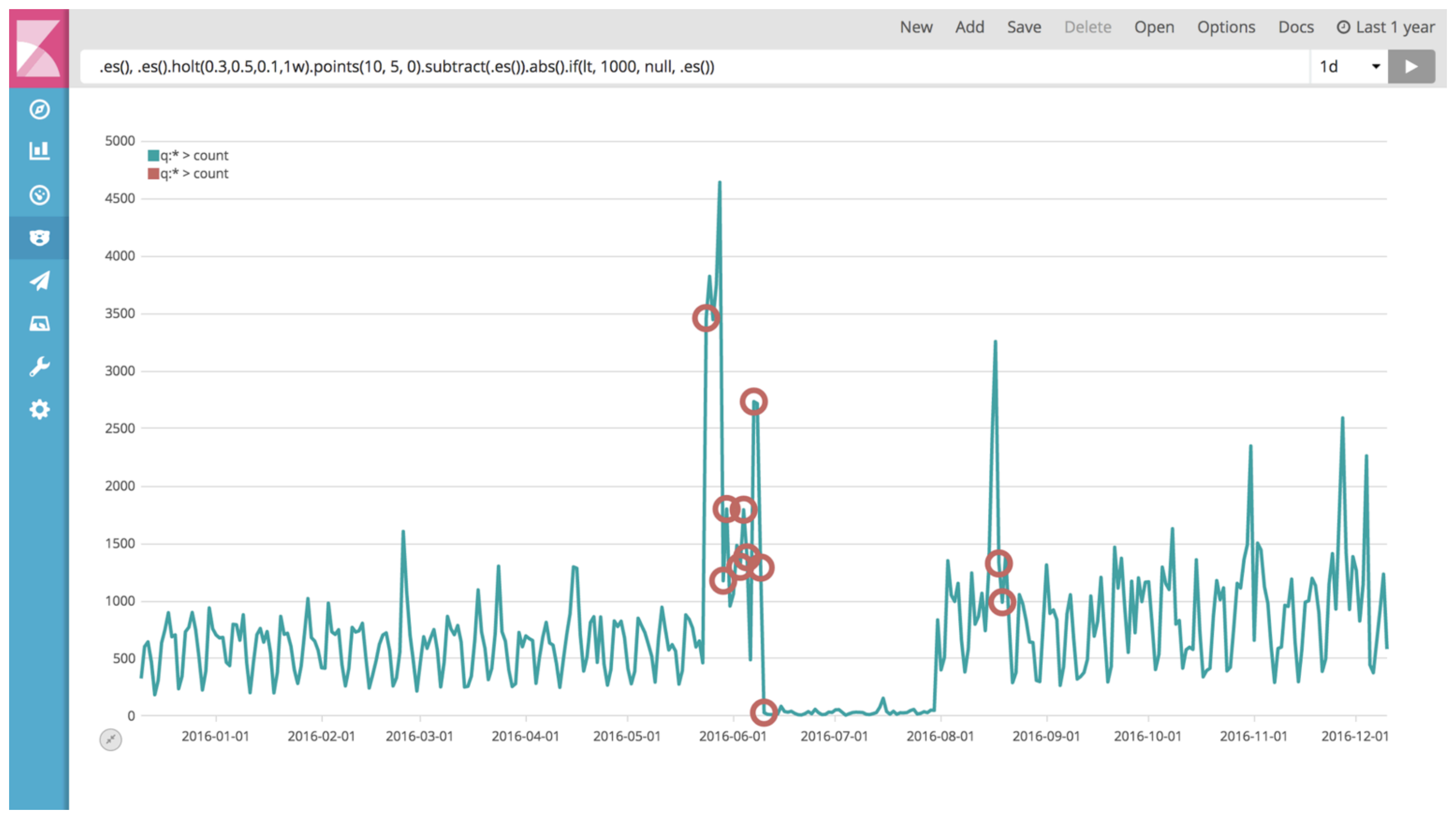
elastic

.if(lt, 1000, **null**, .es())

elastic

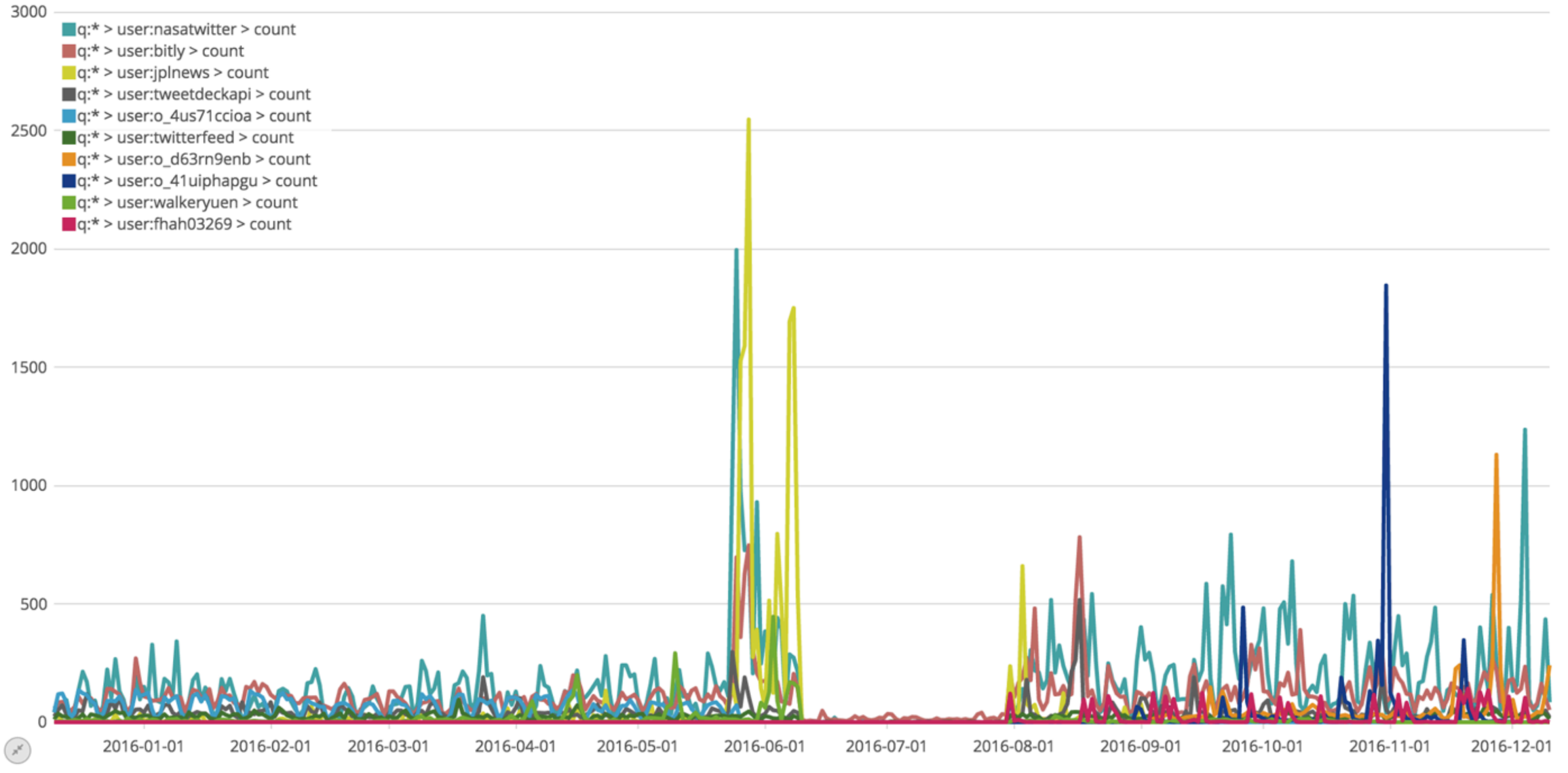
.if(lt, 1000, null, **.es()**)

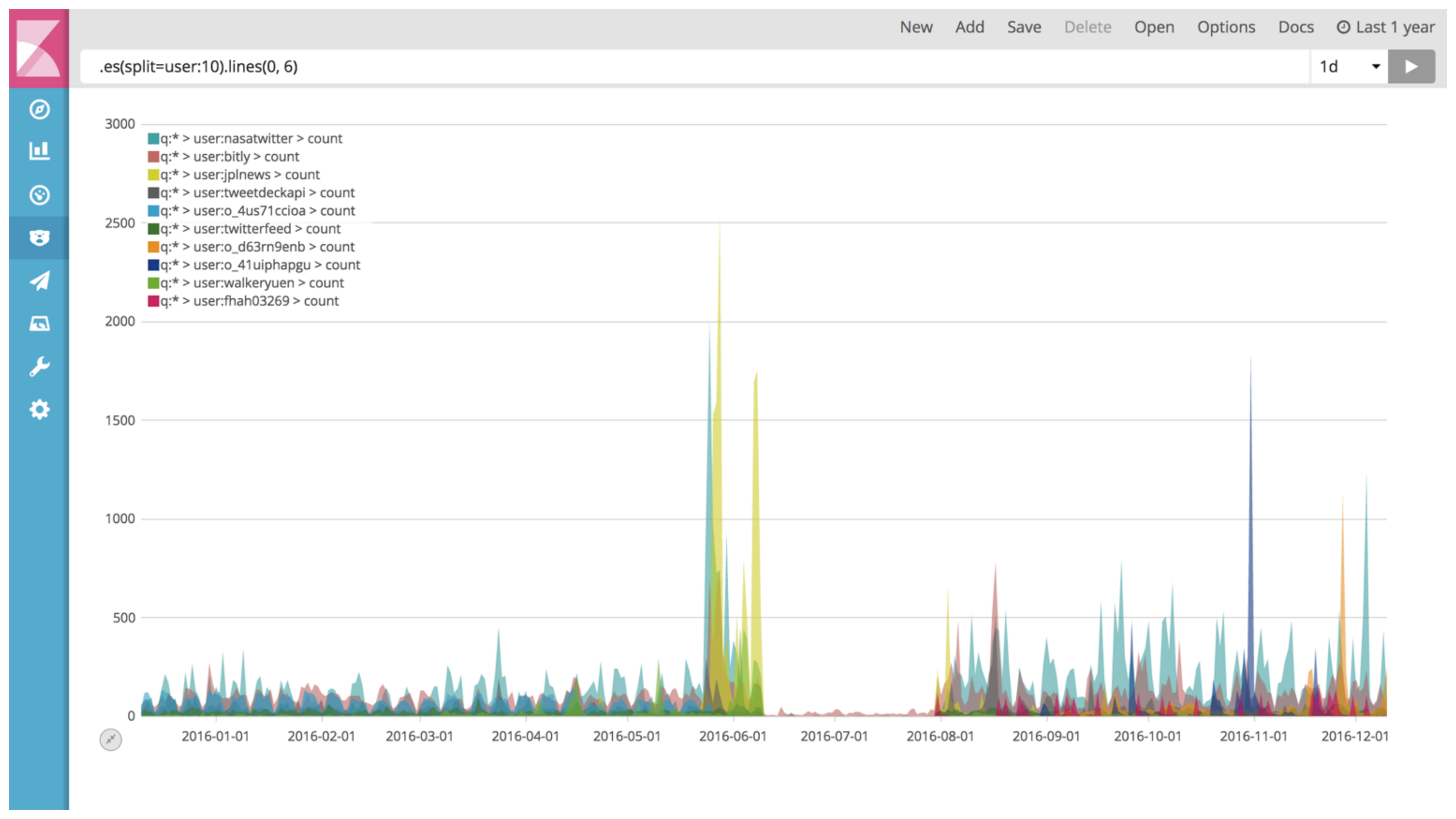
elastic

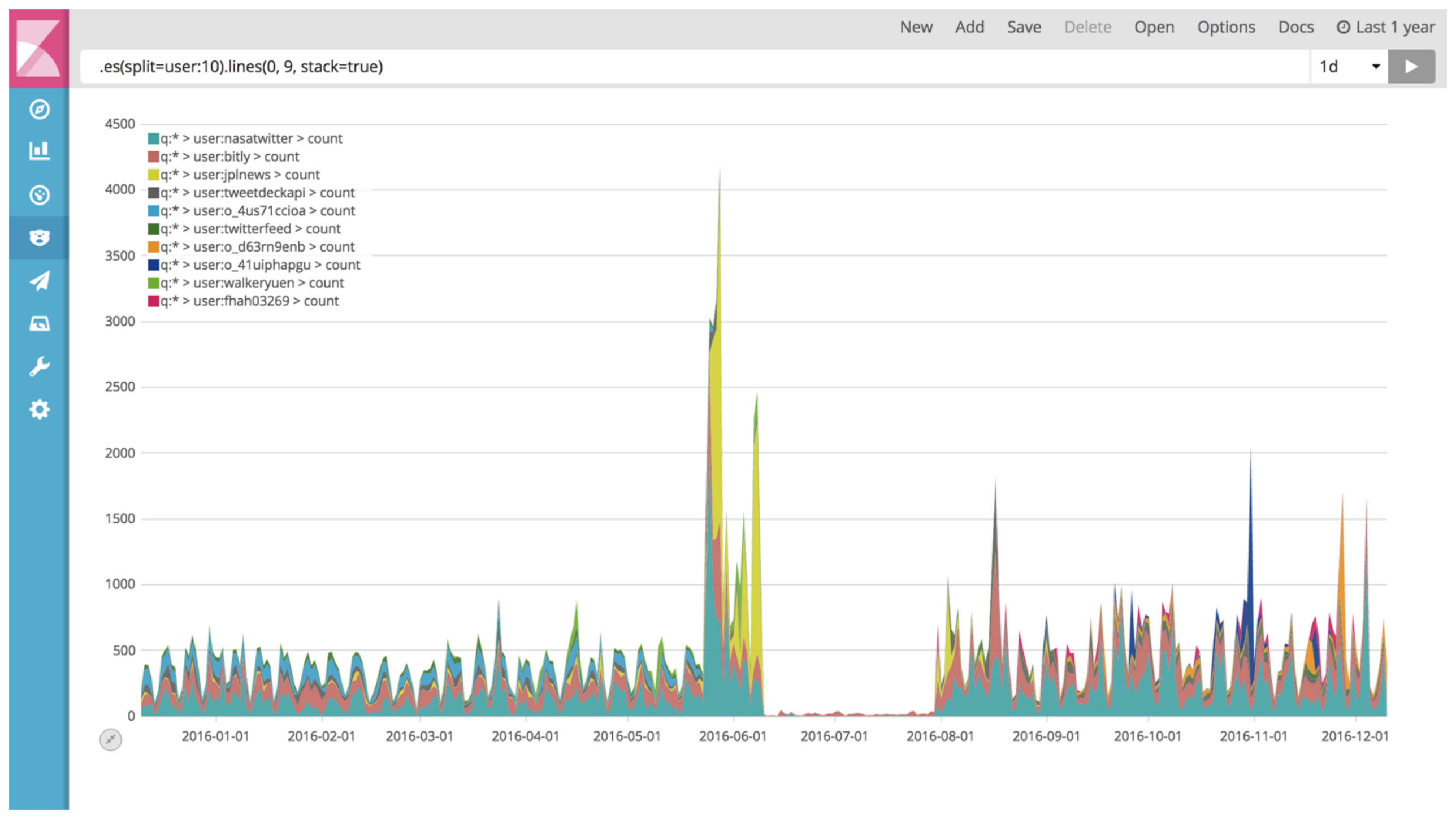

New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(), .es().holt(0.3,0.5,0.1,1w).points(10, 5, 0).subtract(.es()).abs().if(lt, 1000, null, .es())
1d
q:* > count
q:* > count
5000
4500
4000
3500
3000
2500
2000
1500
1000
500
0
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

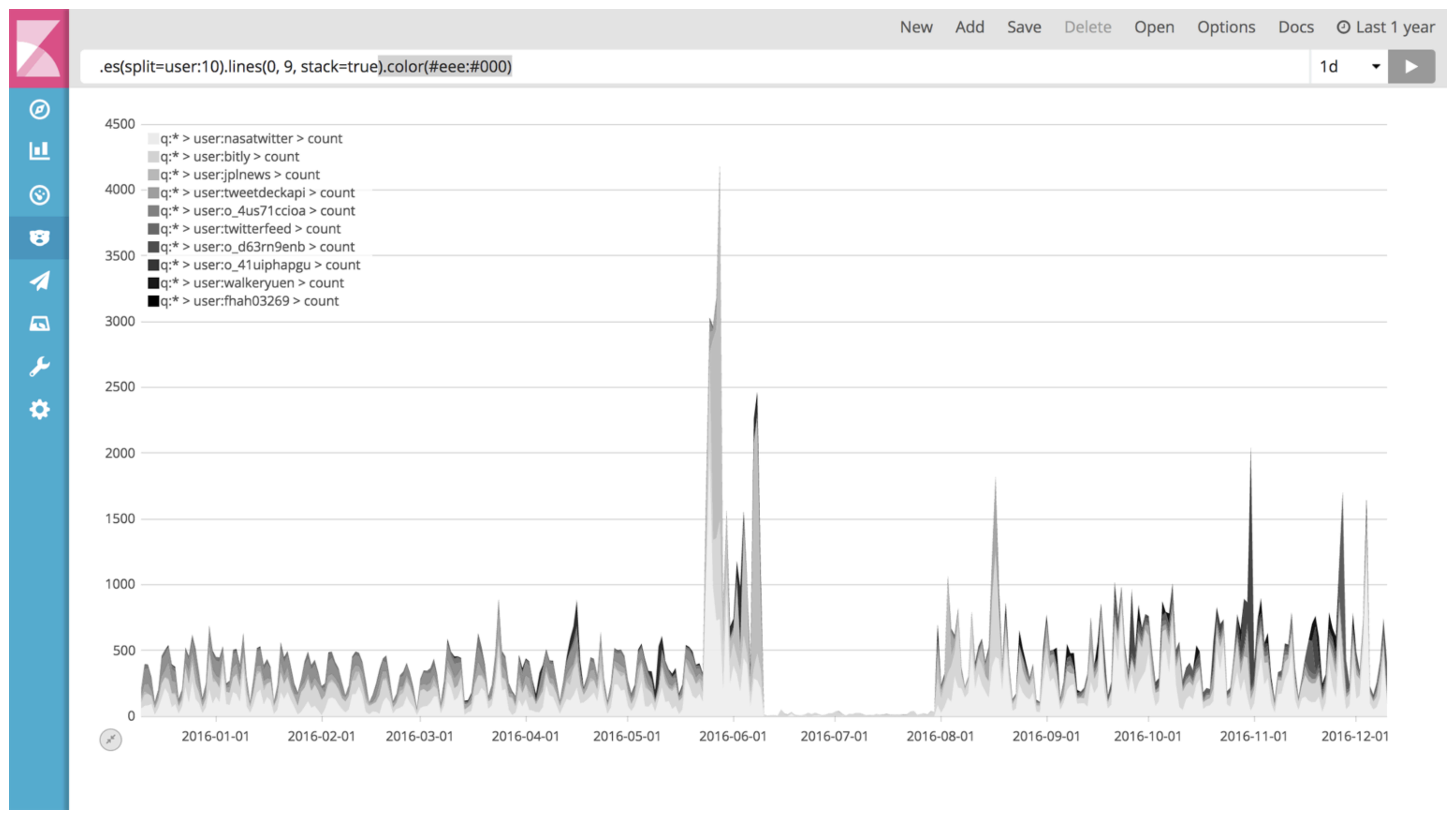
New Add Save Delete Open Options Docs Last 1 year

.es(split=user:10) 1d




New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(split=user:10).lines(0, 6)
1d
3000
2500
2000
1500
1000
500
0
q:* > user:nasatwitter > count
q:* > user:bitly > count
q:* > user:jplnews > count
q:* > user:tweetdeckapi > count
q:* > user:o_4us71ccioa > count
q:* > user:twitterfeed > count
q:* > user:o_d63rn9enb > count
q:* > user:o_41uiphapgu > count
q:* > user:walkeryuen > count
q:* > user:fhah03269 > count
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

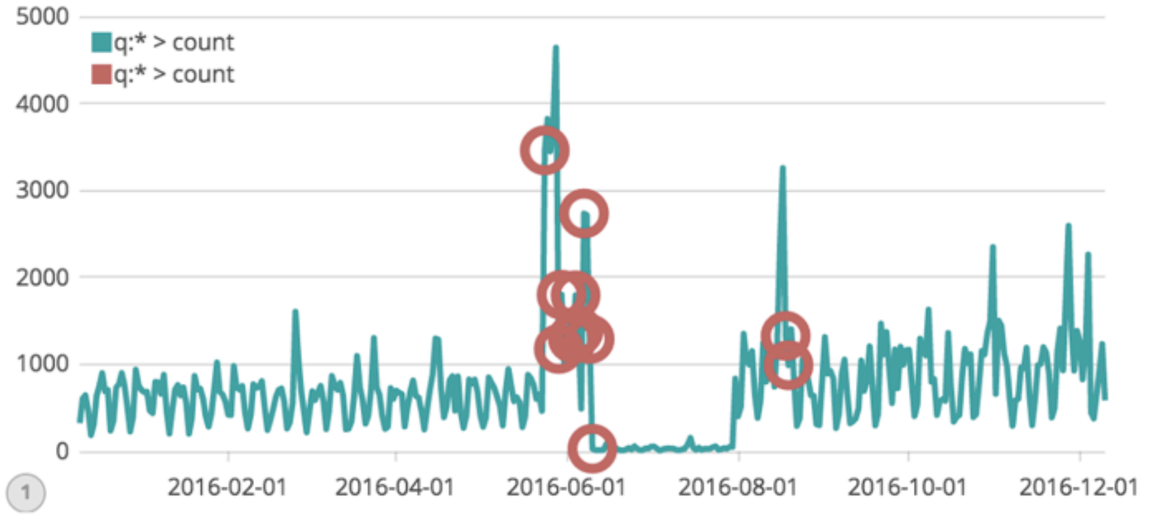

New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(split=user:10).lines(0, 9, stack=true)
1d
q:* > user:nasatwitter > count
q:* > user:bitly > count
q:* > user:jplnews > count
q:* > user:tweetdeckapi > count
q:* > user:o_4us71ccioa > count
q:* > user:twitterfeed > count
q:* > user:o_d63rn9enb > count
q:* > user:o_41uiphapgu > count
q:* > user:walkeryuen > count
q:* > user:fhah03269 > count
4500
4000
3500
3000
2500
2000
1500
1000
500
0
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

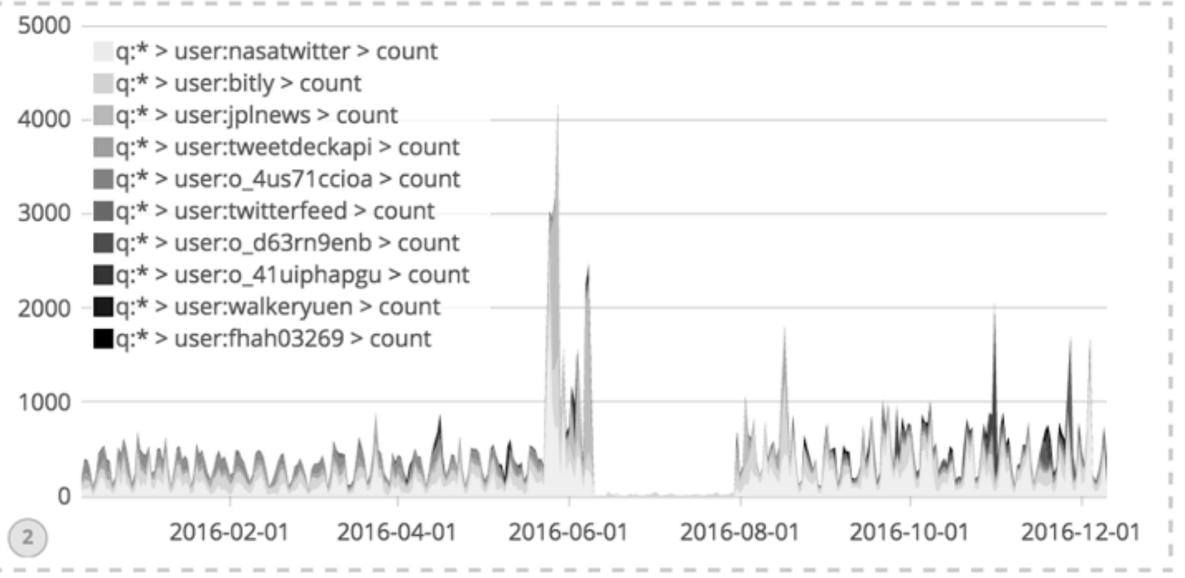

New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(split=user:10).lines(0, 9, stack=true).color(#eee:#000)
1d
q:* > user:nasatwitter > count
q:* > user:bitly > count
q:* > user:jplnews > count
q:* > user:tweetdeckapi > count
q:* > user:o_4us71ccioa > count
q:* > user:twitterfeed > count
q:* > user:o_d63rn9enb > count
q:* > user:o_41uiphapgu > count
q:* > user:walkeryuen > count
q:* > user:fhah03269 > count
4500
4000
3500
3000
2500
2000
1500
1000
500
0
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

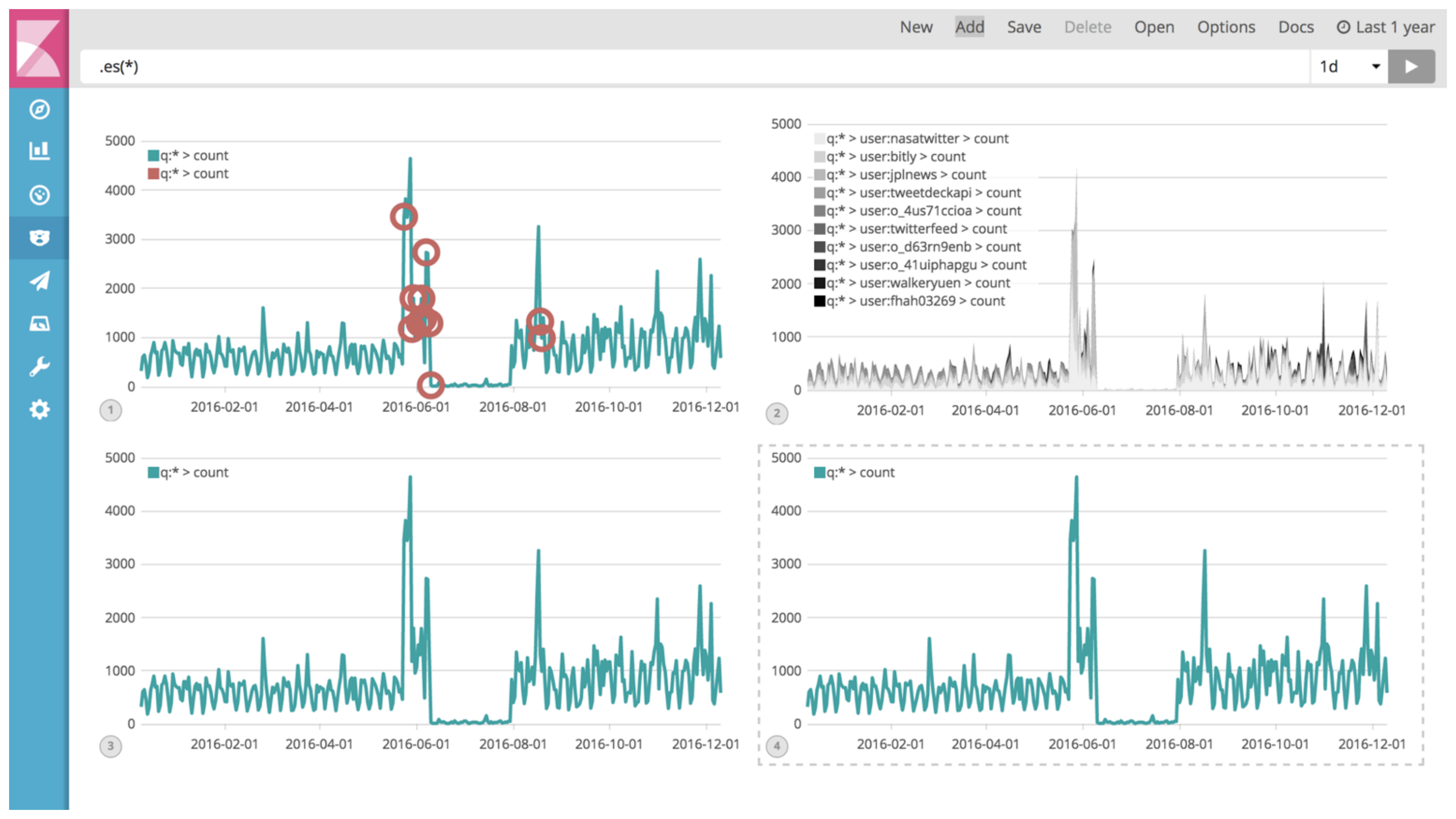
New Add Save Delete Open Options Docs Last 1 year

.es(split=user:10).lines(0, 9, stack=true).color(#eee:#000)

1d



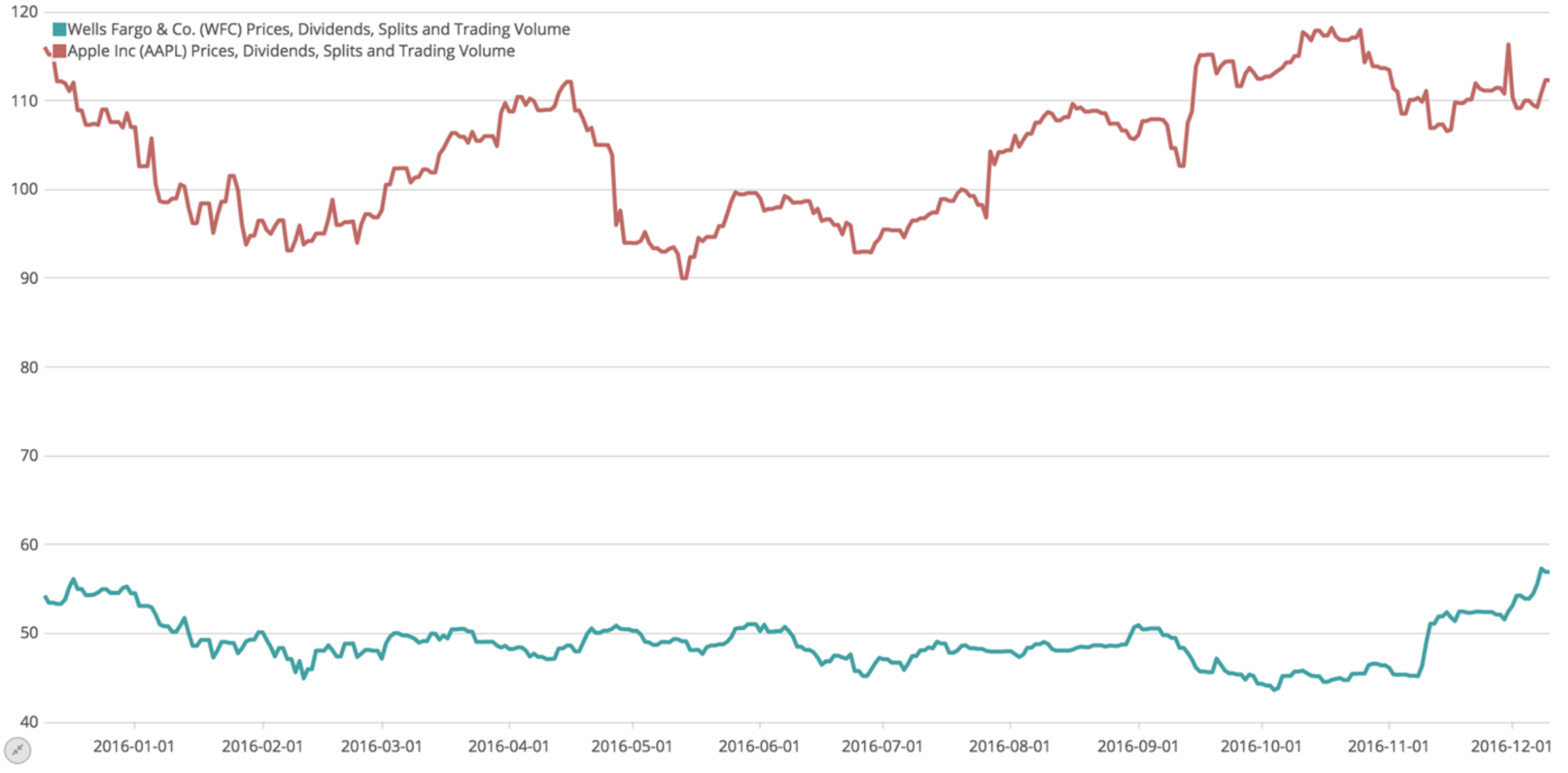


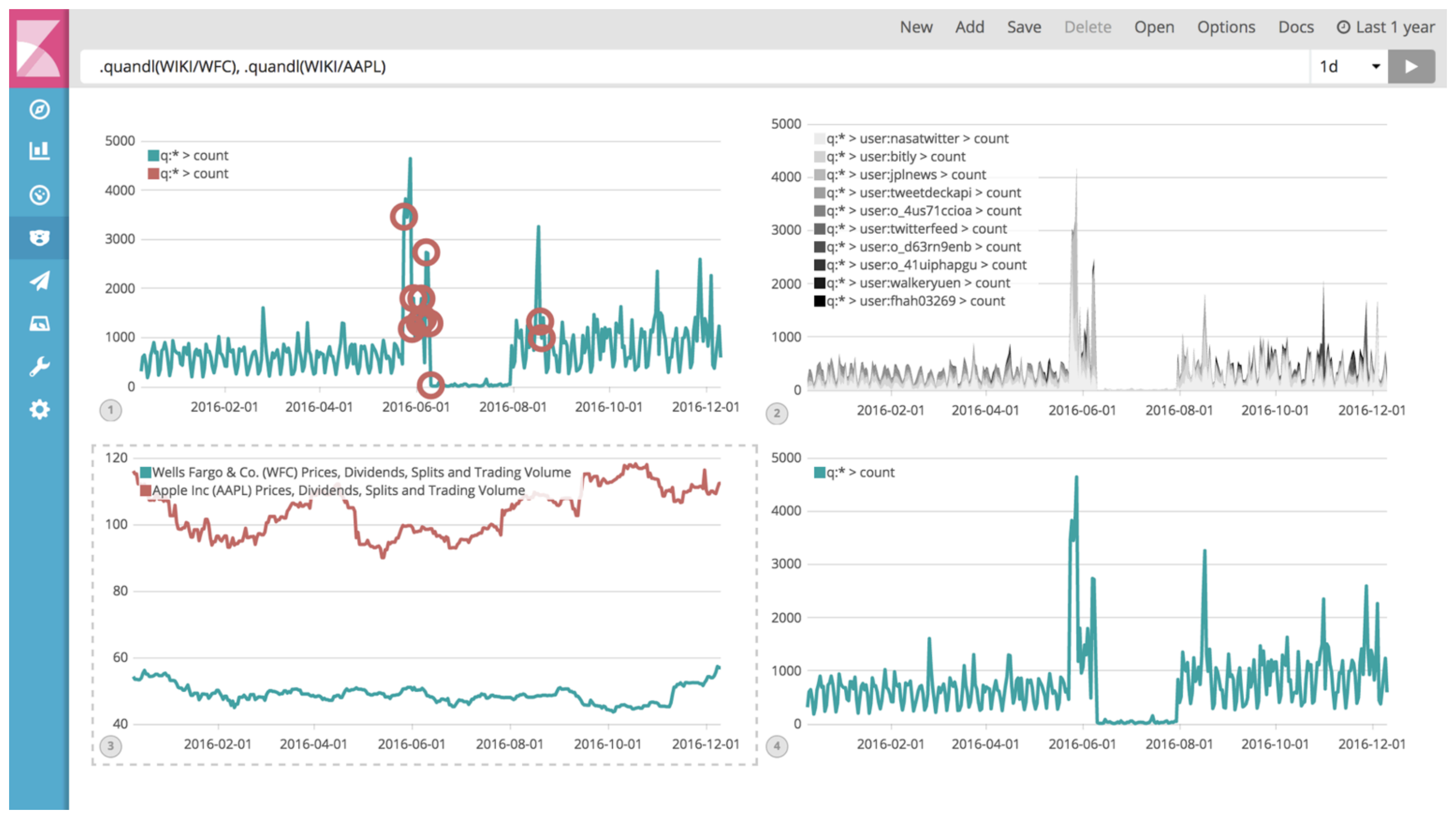

New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(*)
1d
q:* > count
q:* > count
q:* > user:nasatwitter > count
q:* > user:bitly > count
q:* > user:jplnews > count
q:* > user:tweetdeckapi > count
q:* > user:o_4us71ccioa > count
q:* > user:twitterfeed > count
q:* > user:o_d63rn9enb > count
q:* > user:o_41uiphapgu > count
q:* > user:walkeryuen > count
q:* > user:fhah03269 > count
5000
4000
3000
2000
1000
0
2016-02-01
2016-04-01
2016-06-01
2016-08-01
2016-10-01
2016-12-01
1
2
3
4

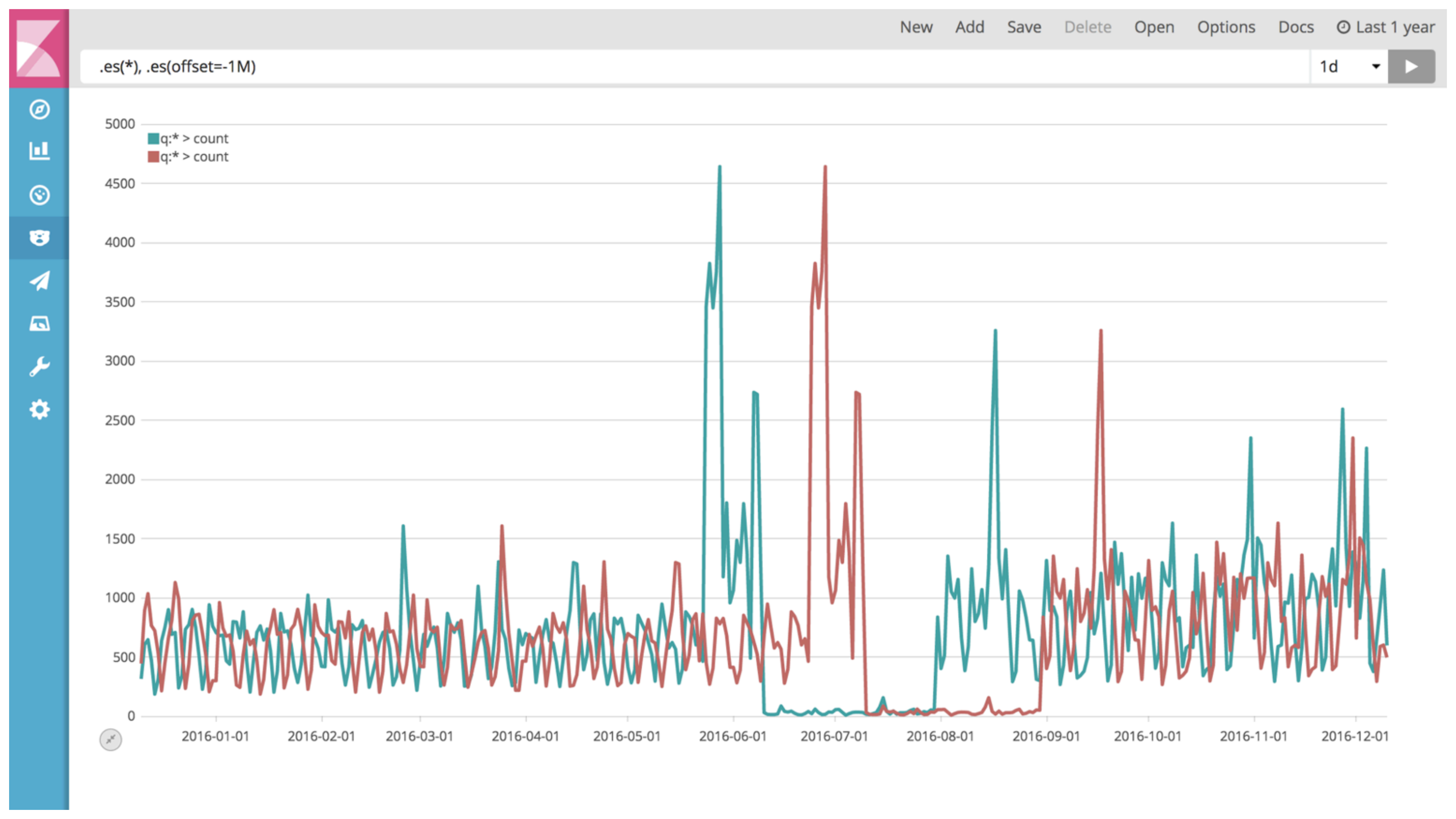
New Add Save Delete Open Options Docs Last 1 year

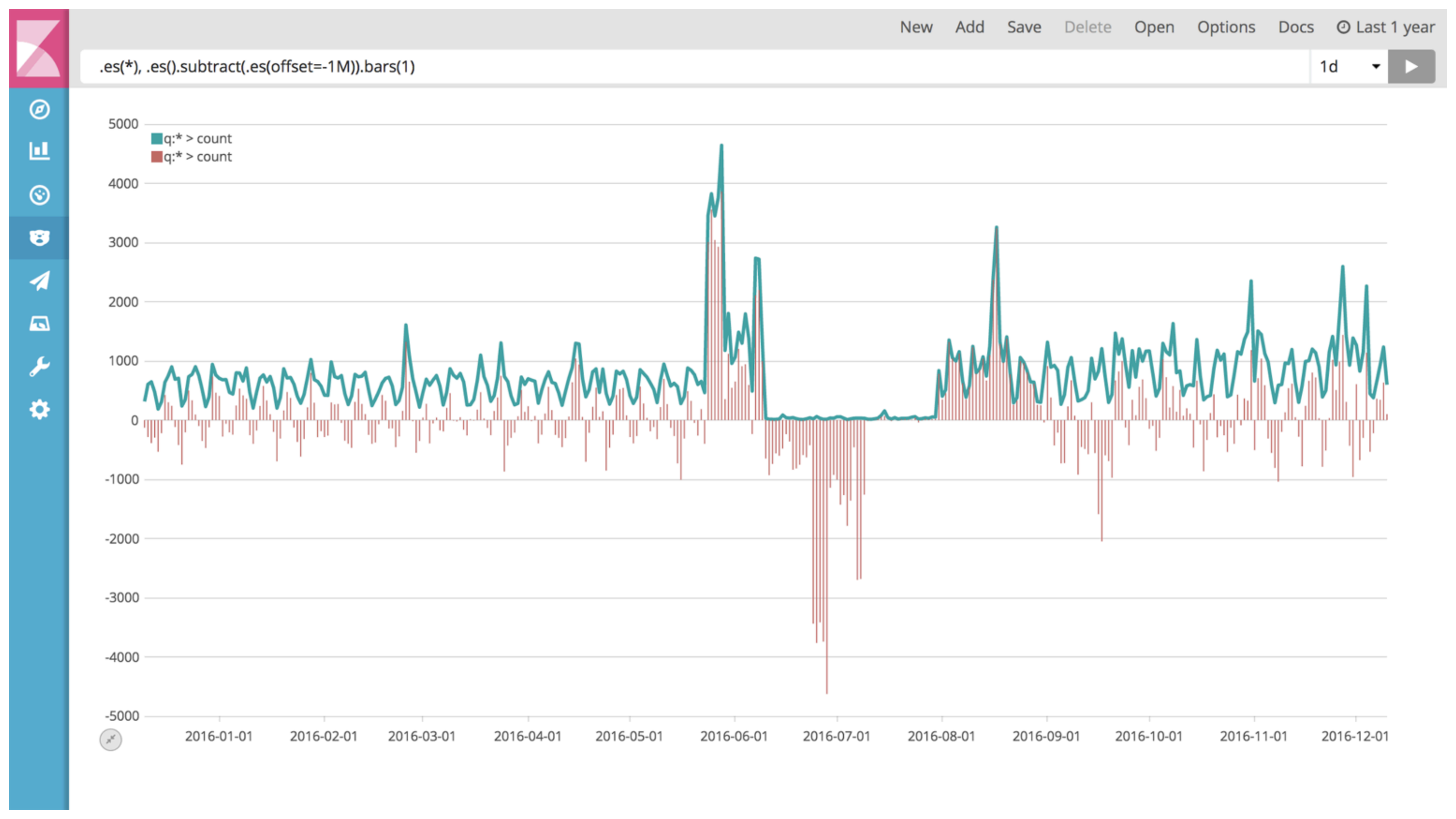
.quandl(WIKI/WFC), .quandl(WIKI/AAPL)

1d



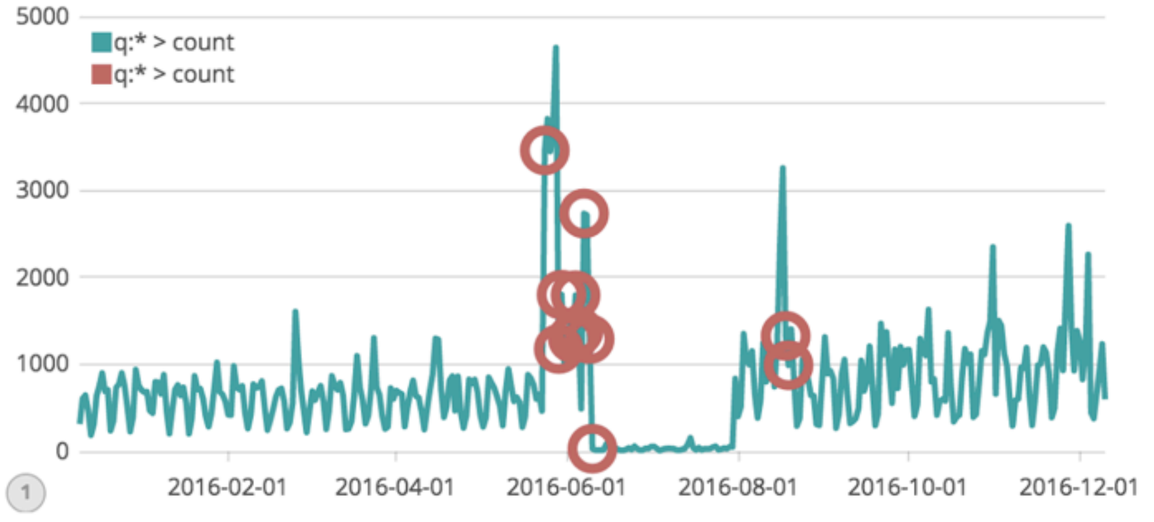

New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.quandl(WIKI/WFC), .quandl(WIKI/AAPL)
1d
q:* > count
q:* > count
5000
4000
3000
2000
1000
0
2016-02-01
2016-04-01
2016-06-01
2016-08-01
2016-10-01
2016-12-01
1
q:* > user:nasatwitter > count
q:* > user:bitly > count
q:* > user:jplnews > count
q:* > user:tweetdeckapi > count
q:* > user:o_4us71ccioa > count
q:* > user:twitterfeed > count
q:* > user:o_d63rn9enb > count
q:* > user:o_41uiphapgu > count
q:* > user:walkeryuen > count
q:* > user:fhah03269 > count
2
Wells Fargo & Co. (WFC) Prices, Dividends, Splits and Trading Volume
Apple Inc (AAPL) Prices, Dividends, Splits and Trading Volume
120
100
80
60
40
3
q:* > count
4

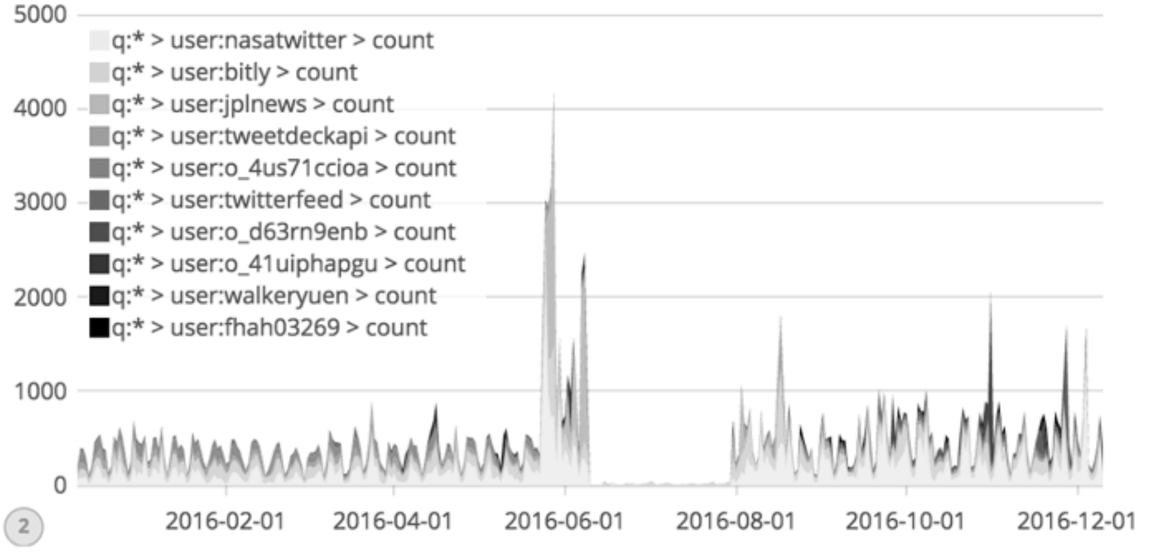

New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(*), .es(offset=-1M)
1d
q:* > count
q:* > count
5000
4500
4000
3500
3000
2500
2000
1500
1000
500
0
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

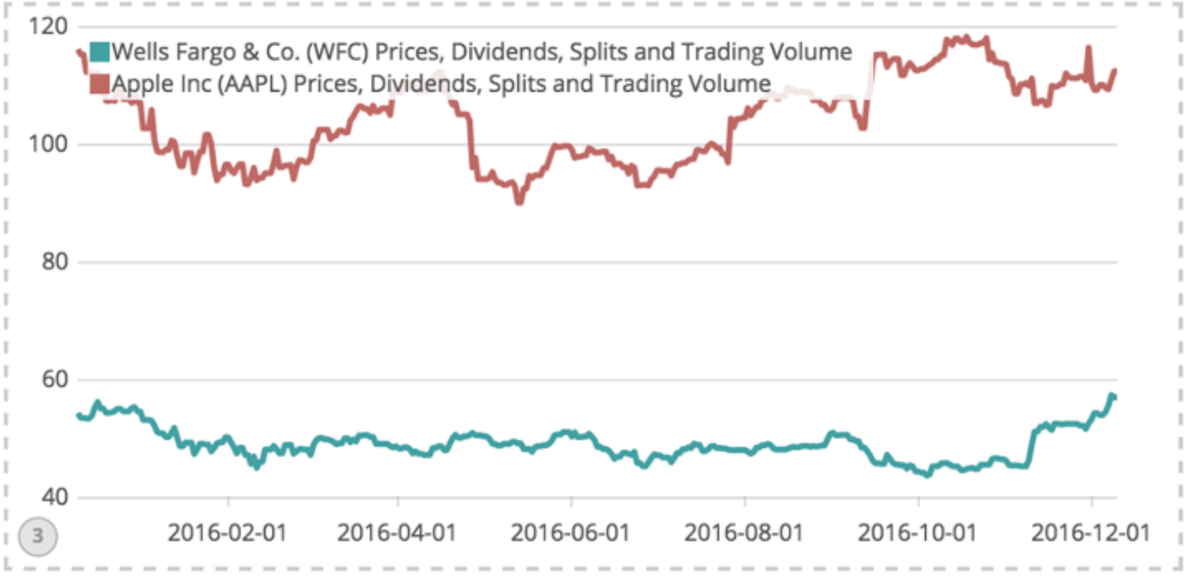

New
Add
Save
Delete
Open
Options
Docs
Last 1 year
.es(*), .es().subtract(.es(offset=-1M)).bars(1)
1d
q:* > count
q:* > count
5000
4000
3000
2000
1000
0
-1000
-2000
-3000
-4000
-5000
2016-01-01
2016-02-01
2016-03-01
2016-04-01
2016-05-01
2016-06-01
2016-07-01
2016-08-01
2016-09-01
2016-10-01
2016-11-01
2016-12-01

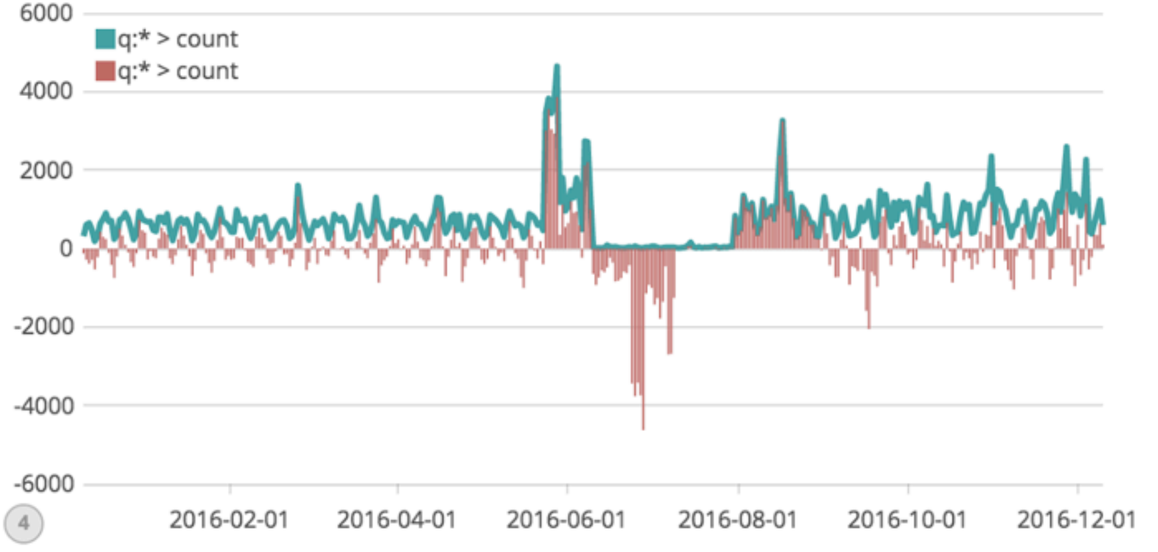
New Add Save Delete Open Options Docs Last 1 year

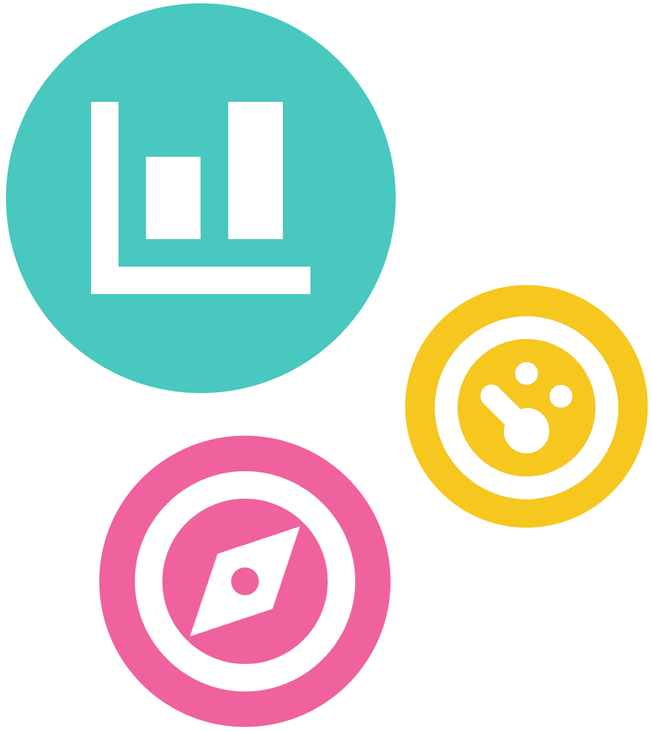
.quandl(WIKI/WFC), .quandl(WIKI/AAPL) 1d



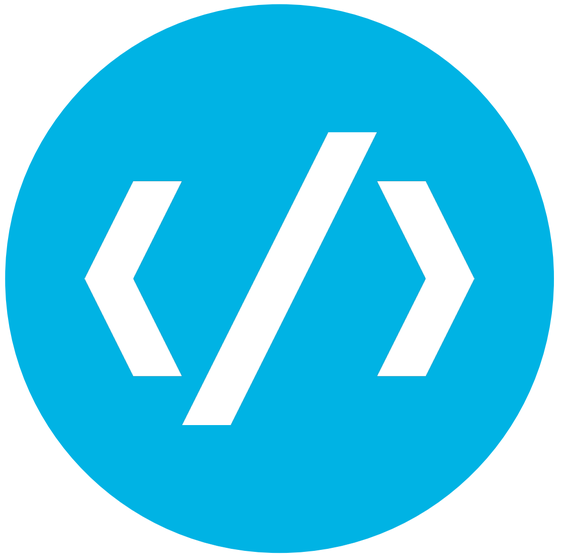








timelion



console

elastic

And more!

https://github.com/rashidkpc/timelion-extras

elastic

Pluggable

https://github.com/rashidkpc/timelion-random

elastic

Kibana 5

That’s what we made

elastic

Kibana 5

What will you make?

elastic