ハンズオンの前に(1/3)

- 本日使用するスライドおよびデータを取得してください

1. jjug_2017_fall.pdf
2. work.zip

- github → https://github.com/higu0shin/JJUG-CCC-fall-2017
- USBメモリ(貸し出しますのでお声かけください)

のいずれか取得してください。

ハンズオンの前に(2/3)

- 下記の3点をダウンロードして解凍する

1. elasticsearch
2. kibana
3. logstash

※3点とも、配布しているworkフォルダの直下に解凍してください

- 公式サイト → https://www.elastic.co/downloads
- USBメモリ(貸し出しますのでお声かけください)

のいずれかから媒体を入手してください。

ハンズオンの前に(3/3)

- それぞれに拡張プラグイン(X-Pack)をインストールする

1.それぞれ解凍したフォルダの下で

"bin/elasticsearch-plugin install x-pack" を実行する

※下線部を"kibana", "logstash"で置き換えてそれぞれ実行する。

2.インターネットに繋がらない場合は、

下記のコマンドでオフラインインストールを実施する

"bin/elasticsearch-plugin install x-pack file:///path/to/x-pack-6.0.0.zip"

Elastic6.0ハンズオン

2017/11/18

#jjug_ccc #ccc_i3

自己紹介

名前 ： 樋口 慎（Acroquest Technology@新横浜）

業務 ： データ分析・検索基盤などの開発や構築支援

Twitter ： @shin0higuchi

コンテンツ

- 概要説明
- データの投入
- ダッシュボードの作成
- Machine Learning
- まとめ

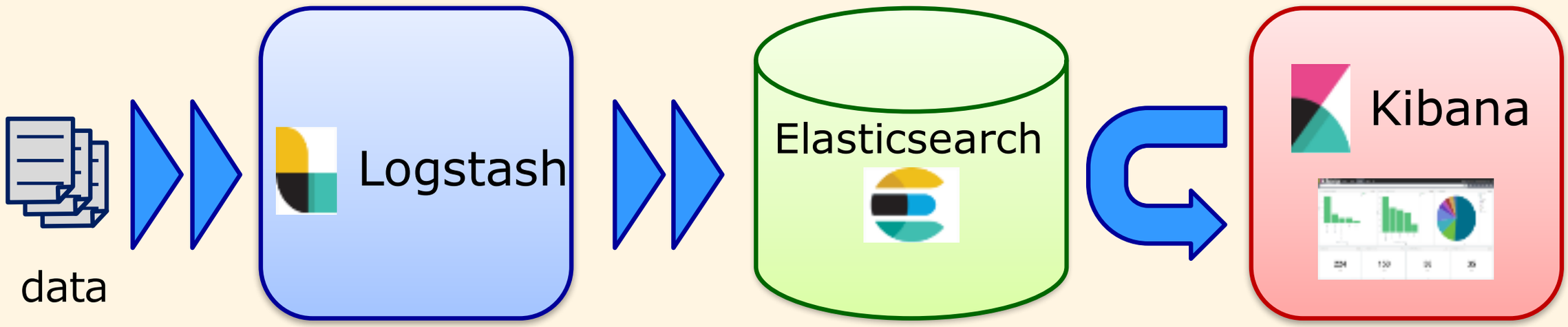
コンテンツ

- 概要説明 ←
- データの投入
- ダッシュボードの作成
- Machine Learning
- まとめ

概要説明

- Elasticsearchとは？
    - ドキュメント指向の全文検索エンジン
    - スケーラビリティが高い
      →高速な検索が可能なため、データ分析などにも活用されている

概要説明



収集・加工・転送　　蓄積・検索　　可視化・分析・管理

コンテンツ

- 概要説明
- データの投入　←
- ダッシュボードの作成
- Machine Learning
- まとめ

データ投入の流れ

1.elasticsearch・kibanaの起動

2.indexの設定(mapping)

3.logstashによるデータ投入

Elasticsearchの起動

- 下記のコマンドでelasticsearchを起動する

  $ bin/elasticsearch
  $ bin\elasticsearch.bat

- パスワードを生成する

  $ bin/x-pack/setup-passwords auto

別ウィンドウorタブを開いて実行

※elastic,kibana,logstash_systemというユーザーのパスワードが表示されるので控えておく

※auto のかわりにinteractiveを使うと手動で設定できる

Kibanaの起動

- config/kibana.ymlを編集する

  + elasticsearch.username: "kibana"
  + elasticsearch.password: "生成されたパスワード"

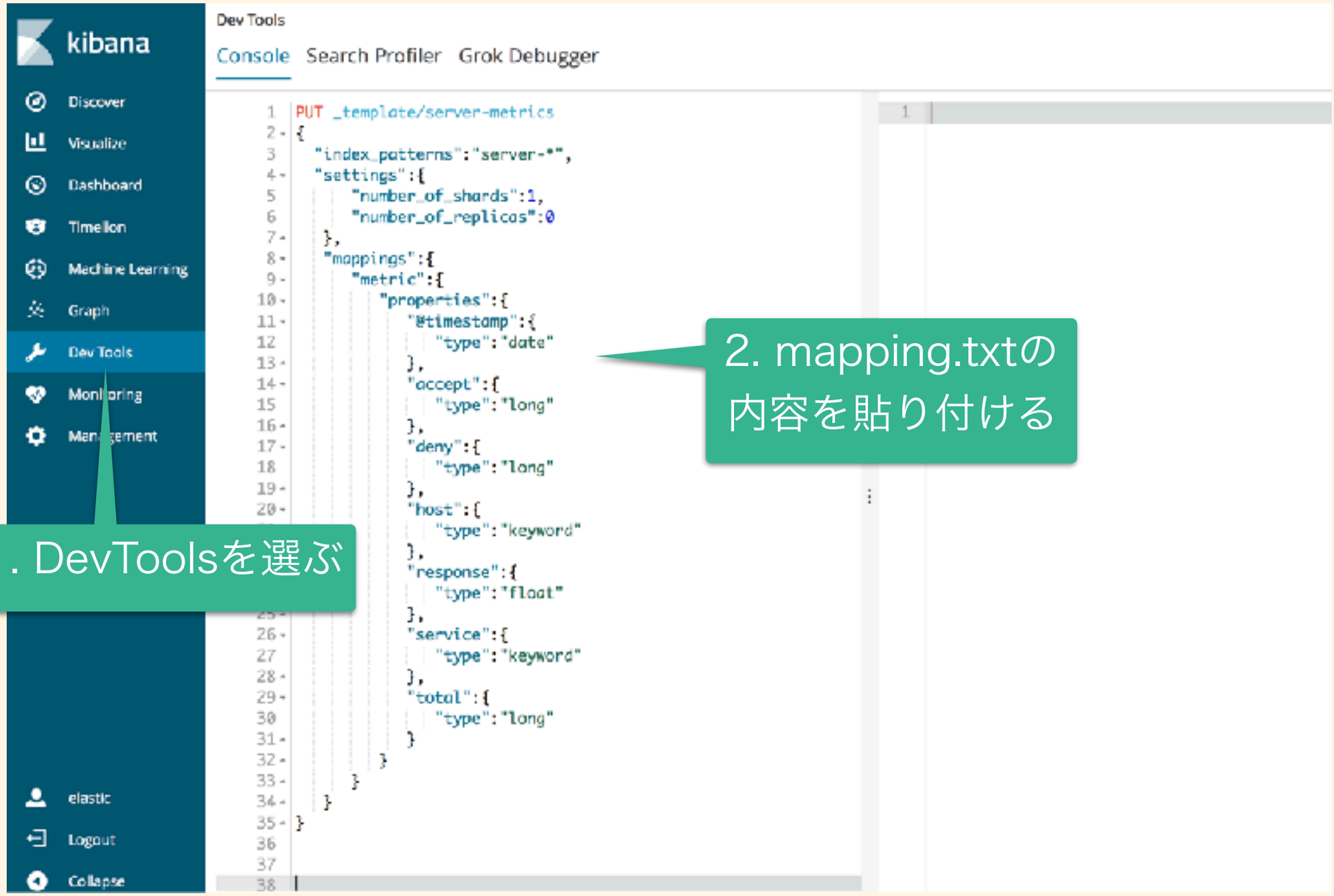
- kibanaを起動する

  $ bin/kibana
  $ bin\kibana.bat

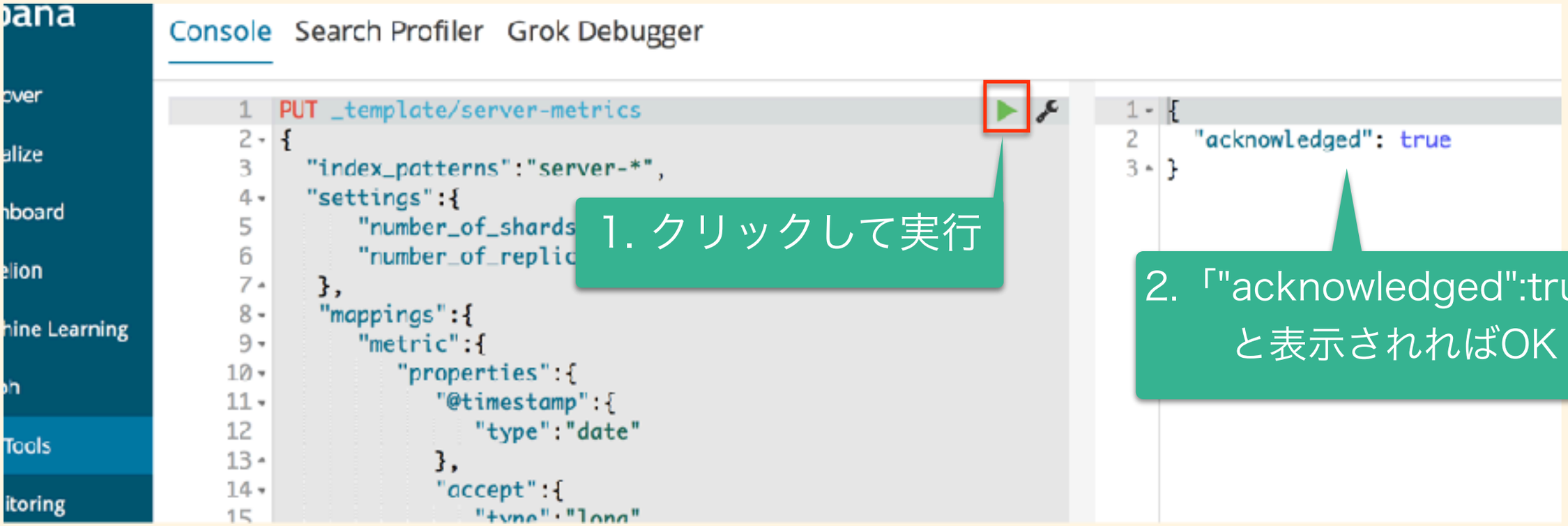
Kibana・Elasticsearchの起動確認

- ブラウザからlocalhost:5601にアクセスする

Mappingの設定


kibana
Discover
Visualize
Dashboard
Timelion
Machine Learning
Graph
Dev Tools
Monitoring
Management
elastic
Logout
Collapse
Dev Tools
Console
Search Profiler
Grok Debugger
PUT _template/server-metrics
{
"index_patterns":"server-*",
"settings":{
"number_of_shards":1,
"number_of_replicas":0
},
"mappings":{
"metric":{
"properties":{
"@timestamp":{
"type":"date"
},
"accept":{
"type":"long"
},
"deny":{
"type":"long"
},
"host":{
"type":"keyword"
},
"response":{
"type":"float"
},
"service":{
"type":"keyword"
},
"total":{
"type":"long"
}
}
}
}
}
1. DevToolsを選ぶ
2. mapping.txtの
内容を貼り付ける

Mappingの設定


Console
Search Profiler
Grok Debugger
PUT _template/server-metrics
{
"index_patterns":"server-*",
"settings":{
"number_of_shards
"number_of_replic
},
"mappings":{
"metric":{
"properties":{
"@timestamp":{
"type":"date"
},
"accept":{
{
"acknowledged": true
}
1. クリックして実行
2. 「"acknowledged":true」
と表示されればOK

Logstashの設定・実行

- config/logstash.ymlを編集する

  + xpack.monitoring.elasticsearch.url: ["localhost:9200"]
  + xpack.monitoring.elasticsearch.username: "logstash_system"
  + xpack.monitoring.elasticsearch.password: "**changeme**"

自分の環境で設定した、
logstash_systemユーザー
のパスワード

Logstashの設定・実行

- server_metrics.confを編集する(input)

```
input{
  file{
    path => "/Users/SHIN/jjug_2017_fall/work/server_metrics/*"
    start_position => "beginning"
    sincedb_path => "/dev/null"    ←Windowsの方は "nul"に書き換える
  }
}
```

Logstashの設定・実行

- server_metrics.confを編集する(output)

```
output{
  stdout{省略}
  elasticsearch{
    hosts => "localhost:9200"
    user => "elastic"
    password => 'P6kHQ$&sj-Aoo^pj1B16'
    index => "server-metrics"
    document_type => "metric"
  }
}
```

自分の環境で設定した、
elasticユーザーのパスワード

Logstashの設定・実行

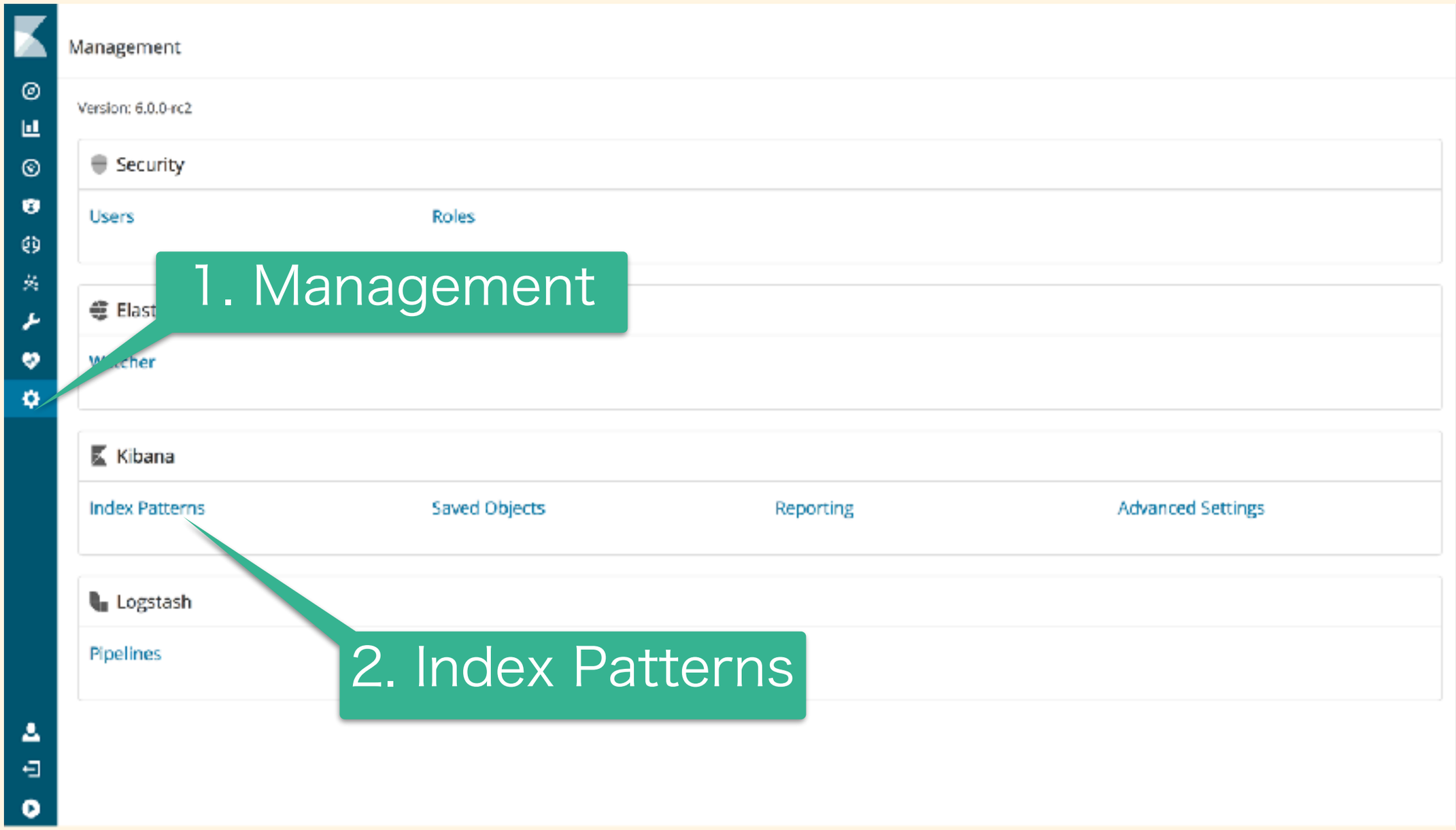
- 下記のコマンドでlogstashを実行

  $ bin/logstash -f ../server_metrics.conf

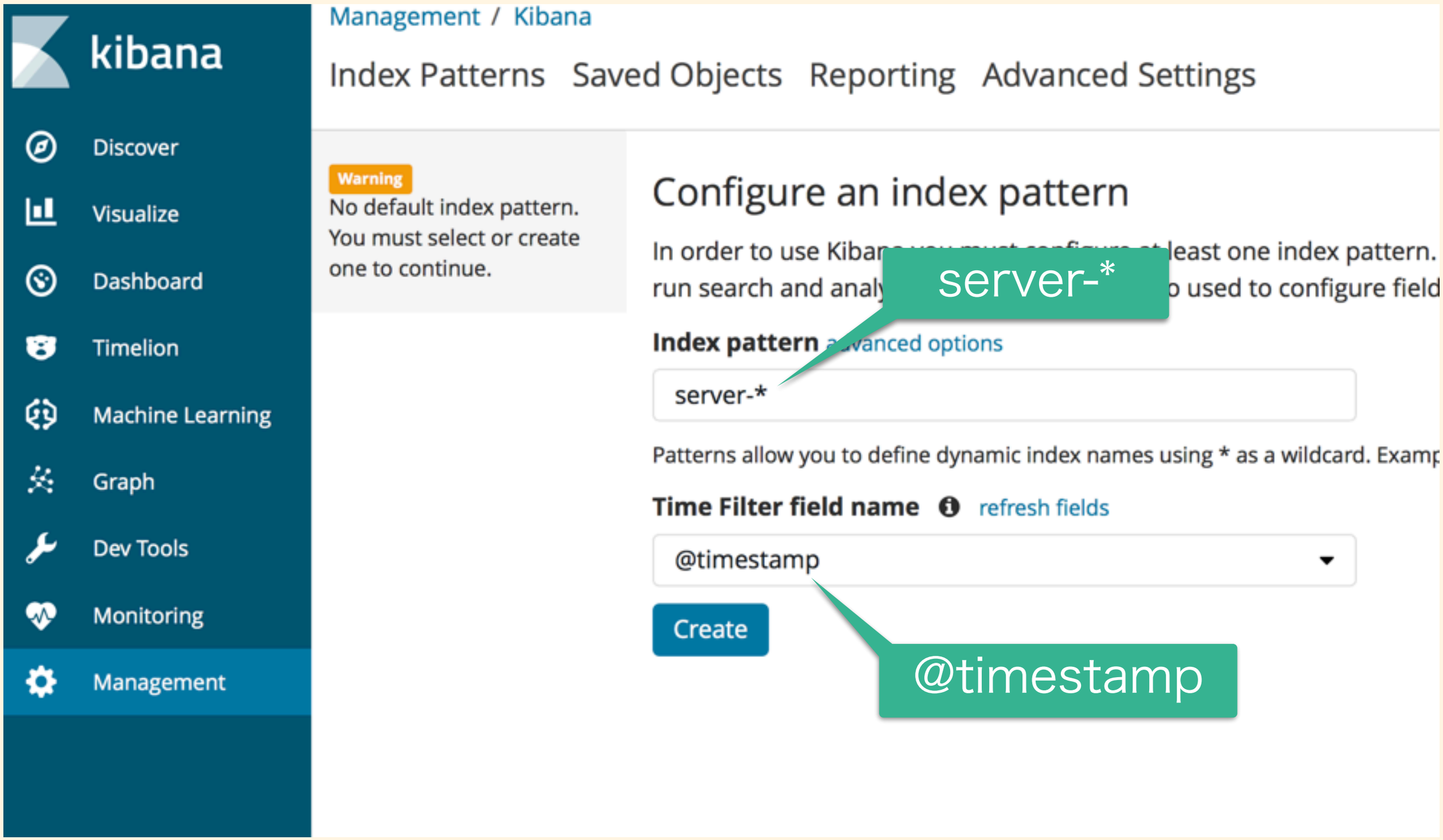
  $ bin\logstash.bat -f ..\server_metrics.conf

  ※実行するとドットが多数出力されます

index patternの作成



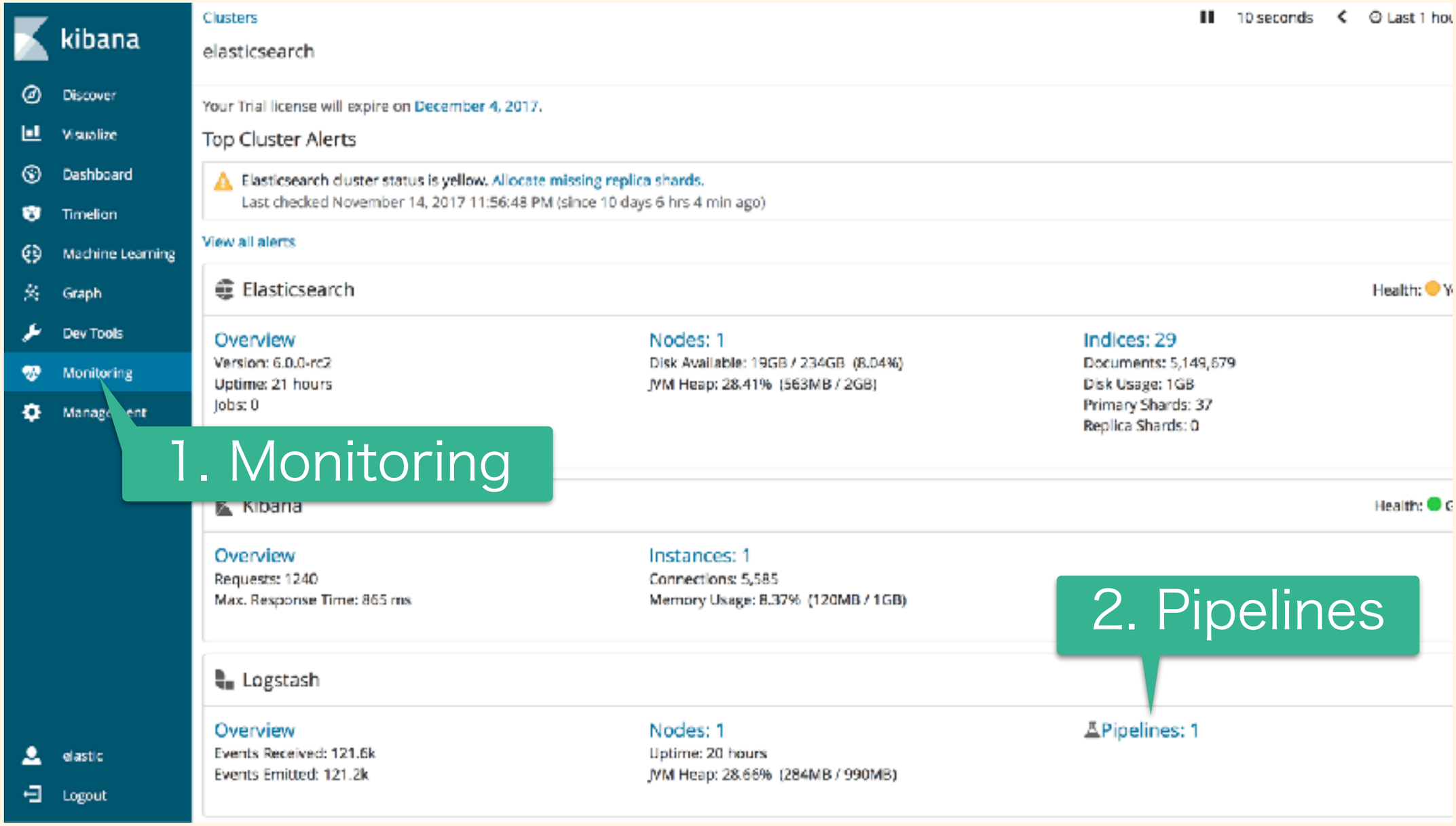
index patternの作成



logstash pipelines

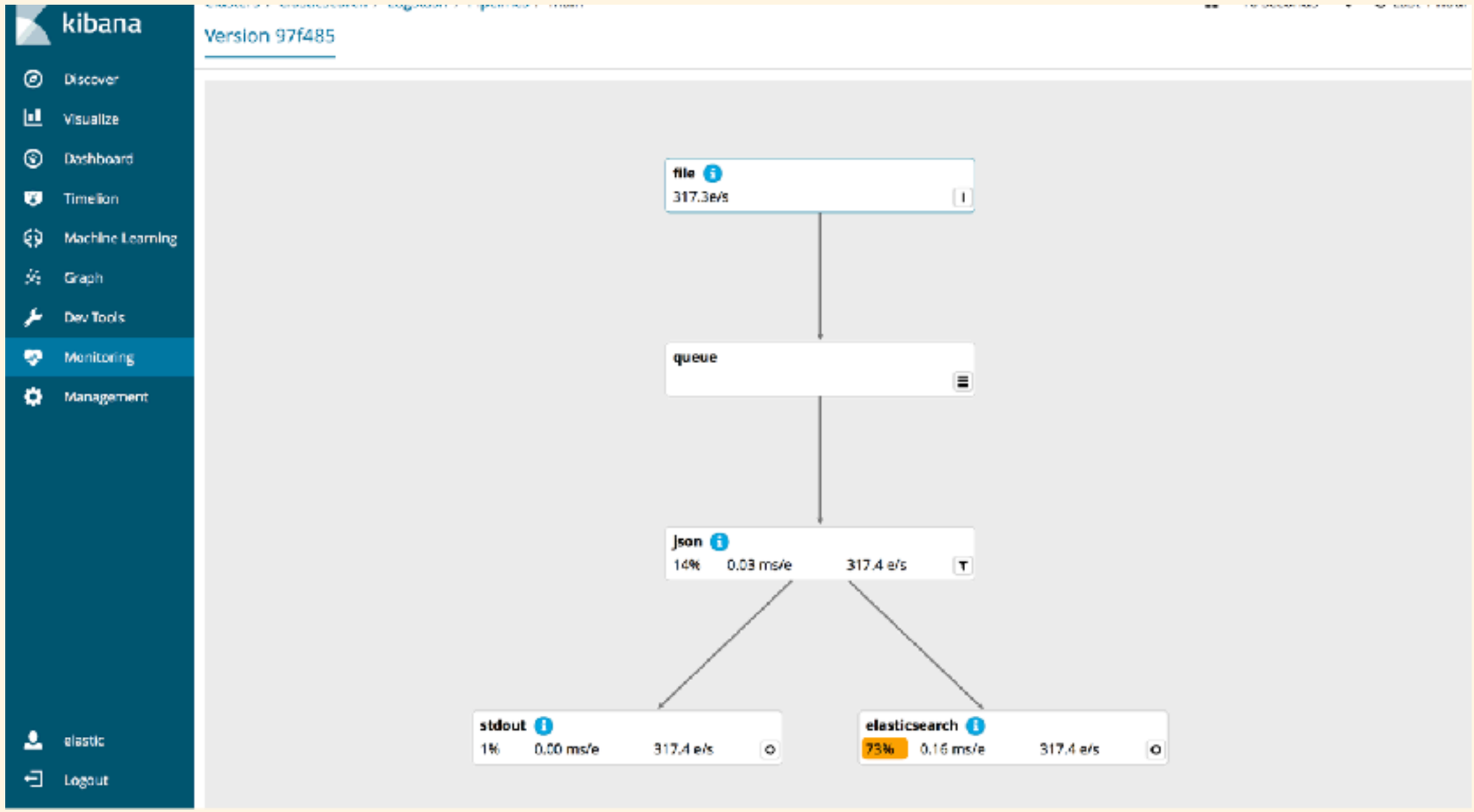
- logstashのpipelineごとに、
  データの流れやスループットをKibana上で確認することができる。

logstash pipelines



logstash pipelines

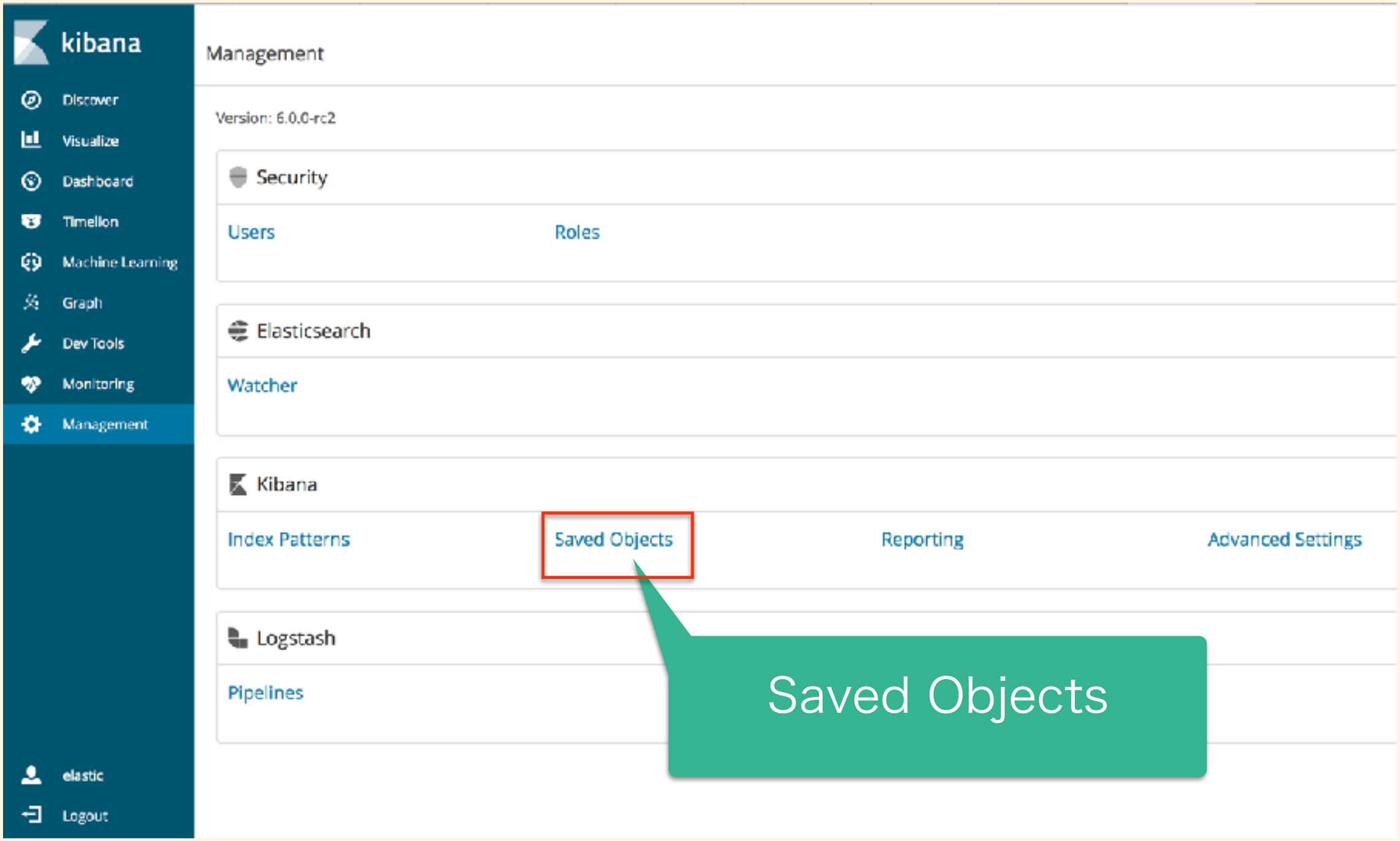
- logstash pipelineの状態が確認できる



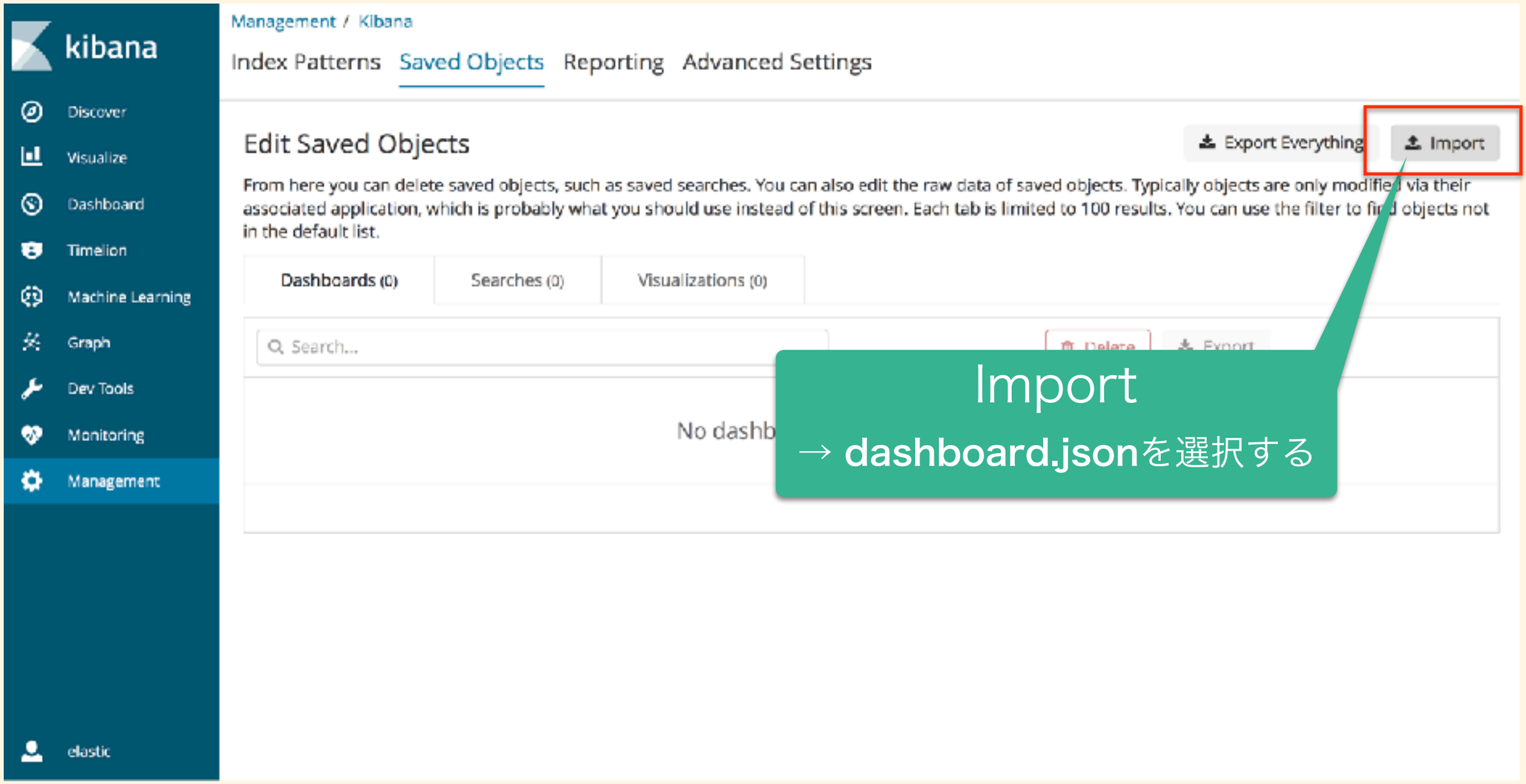
コンテンツ

- 概要説明
- データの投入
- ダッシュボードの作成　←
- Machine Learning
- まとめ

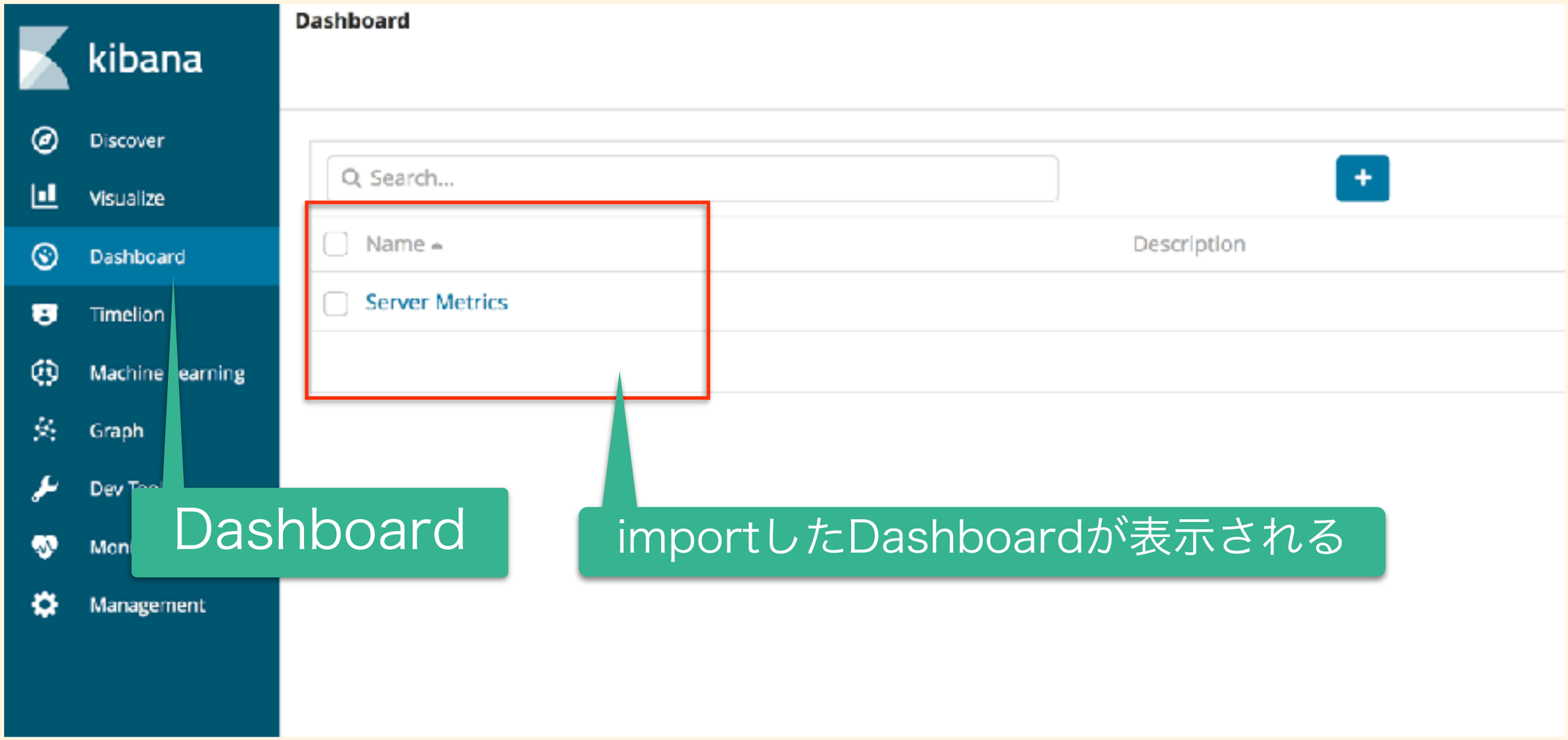
Dashboardのインポート



Dashboardのインポート



Dashboardのインポート



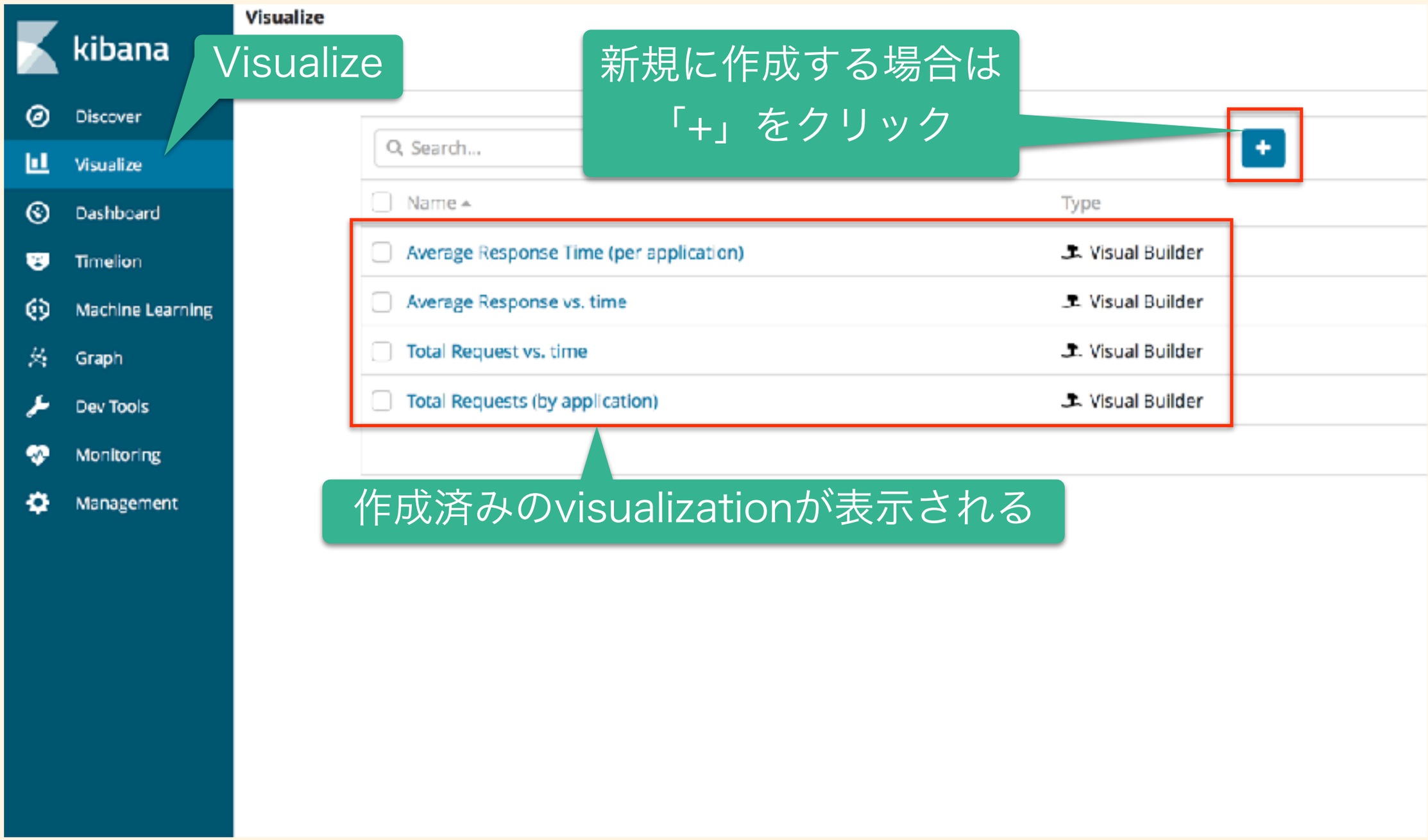
Dashboardのインポート

Dashboardの作成方法

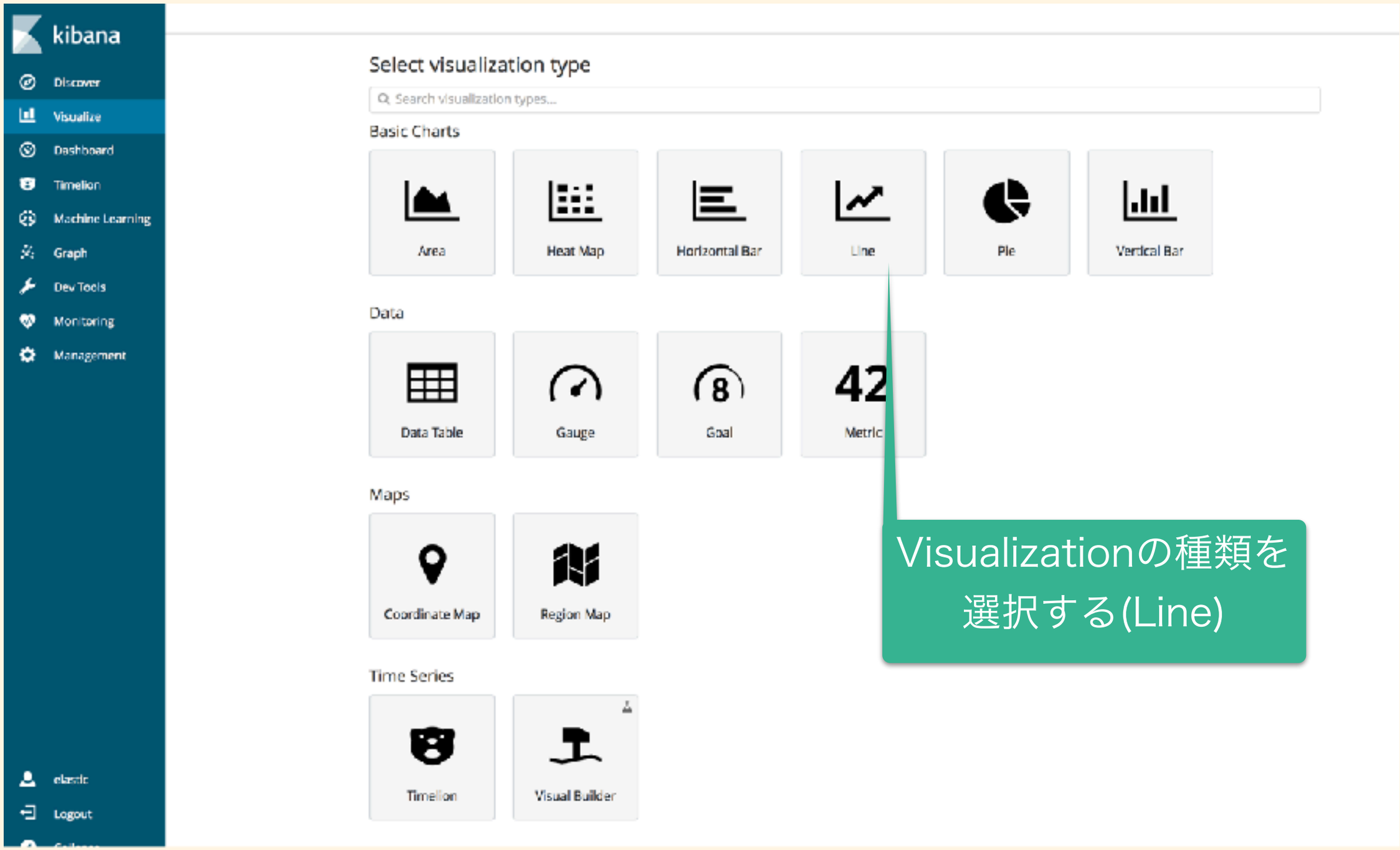
1.Dashboardに配置するvisualizationを作成する

2.Dahboard画面からvisualizationを配置する

Visualizationの作成

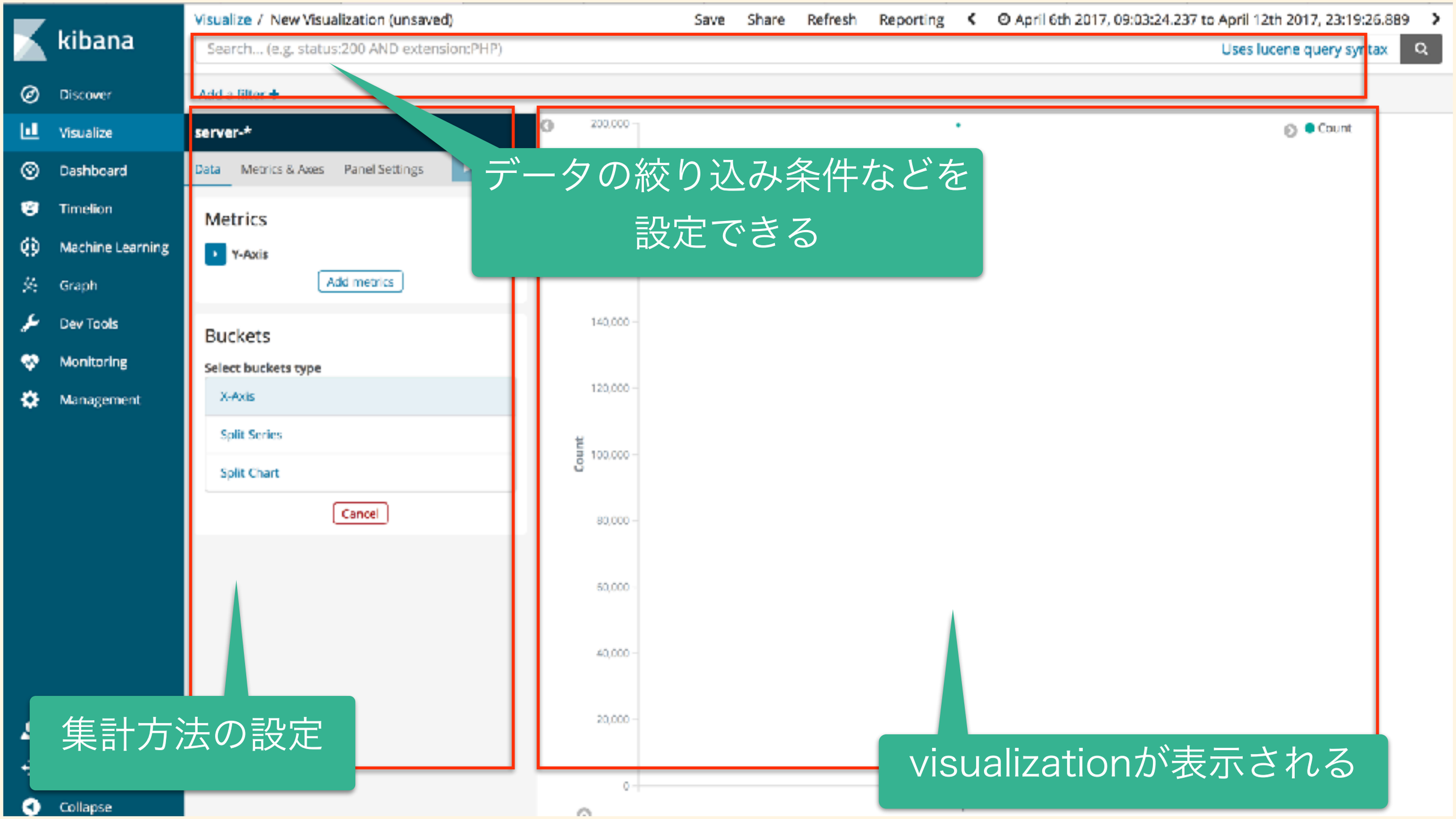


Visualizationの作成

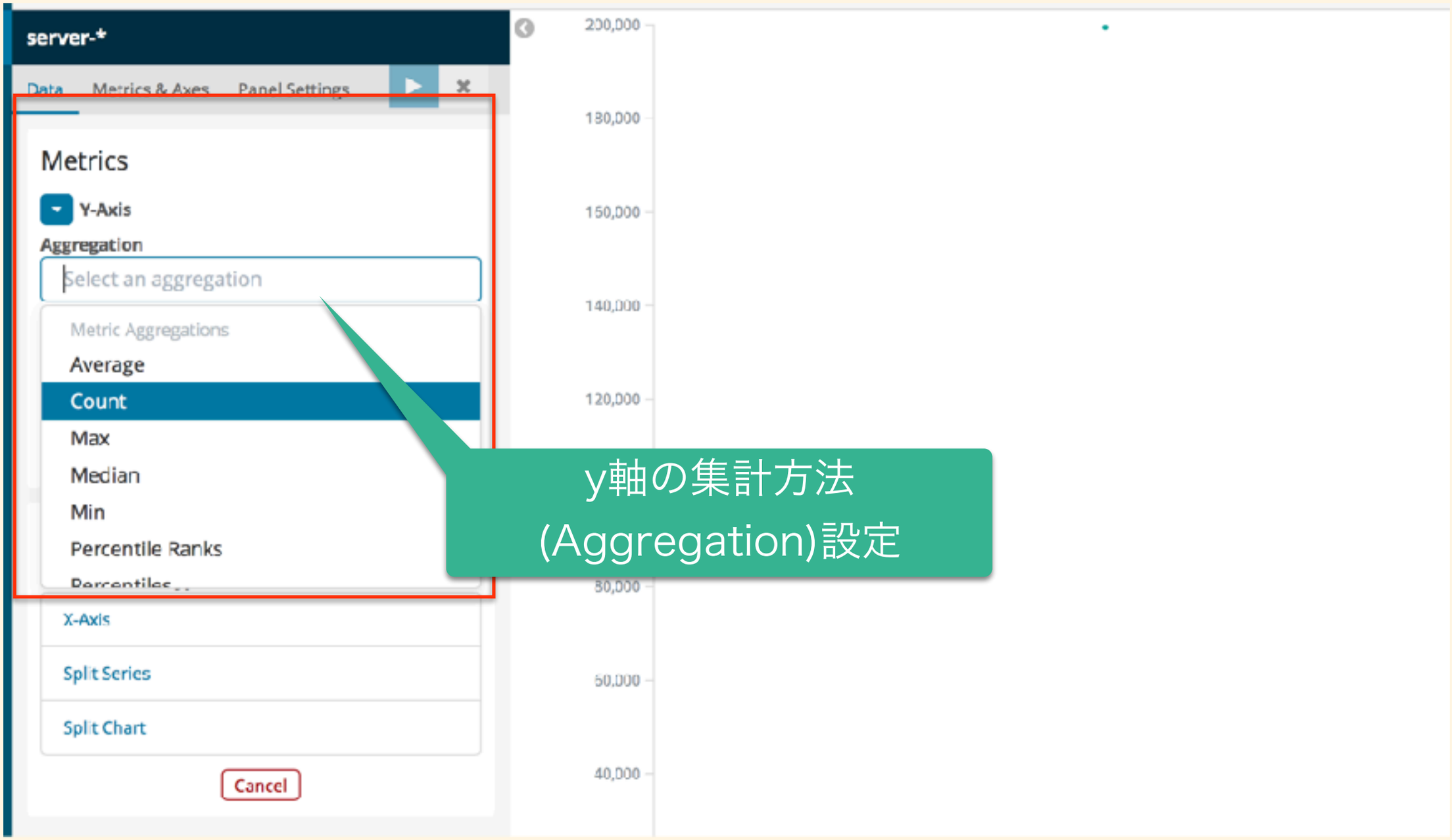


Visualizationの作成

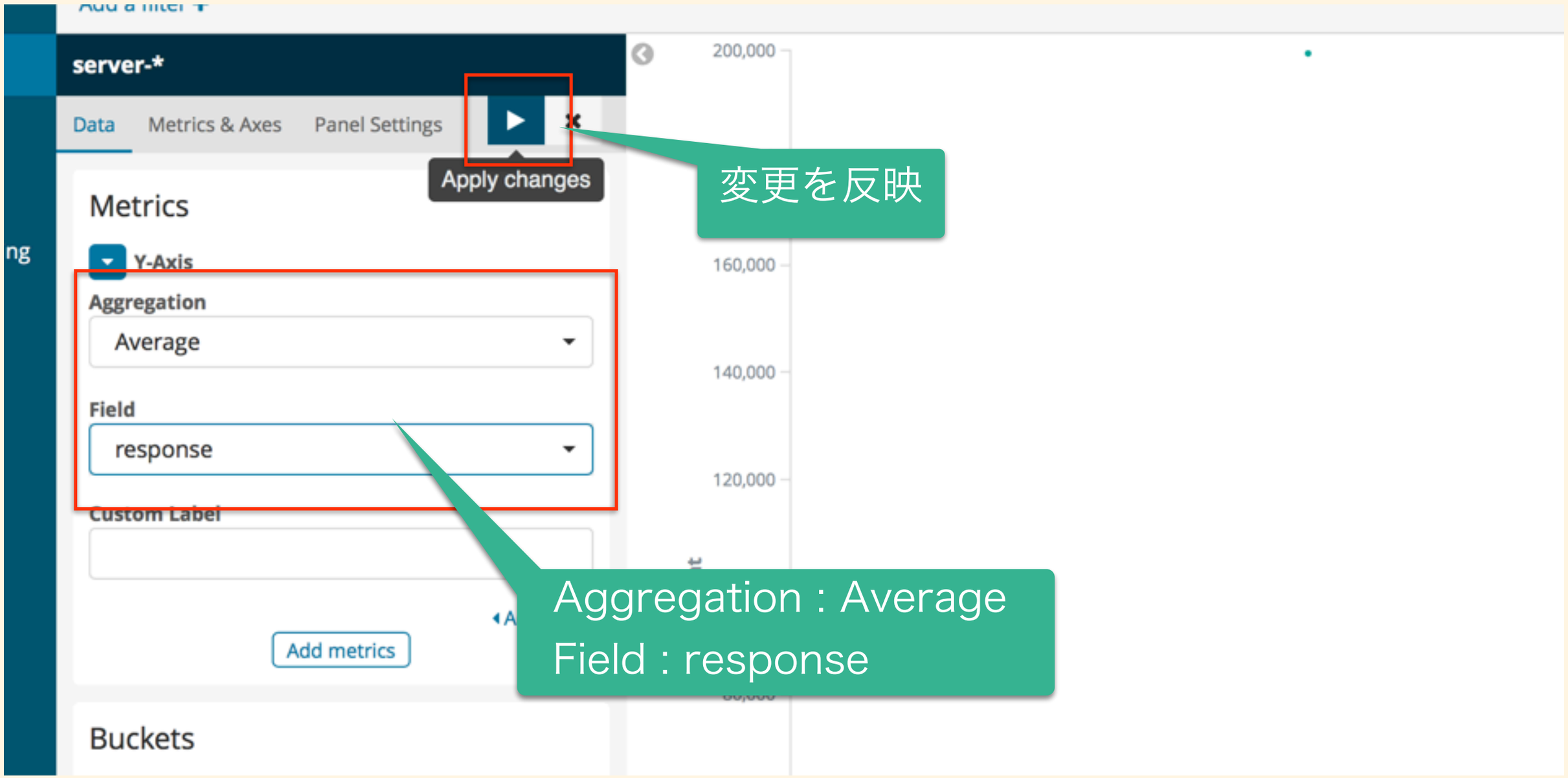
Visualizationの作成



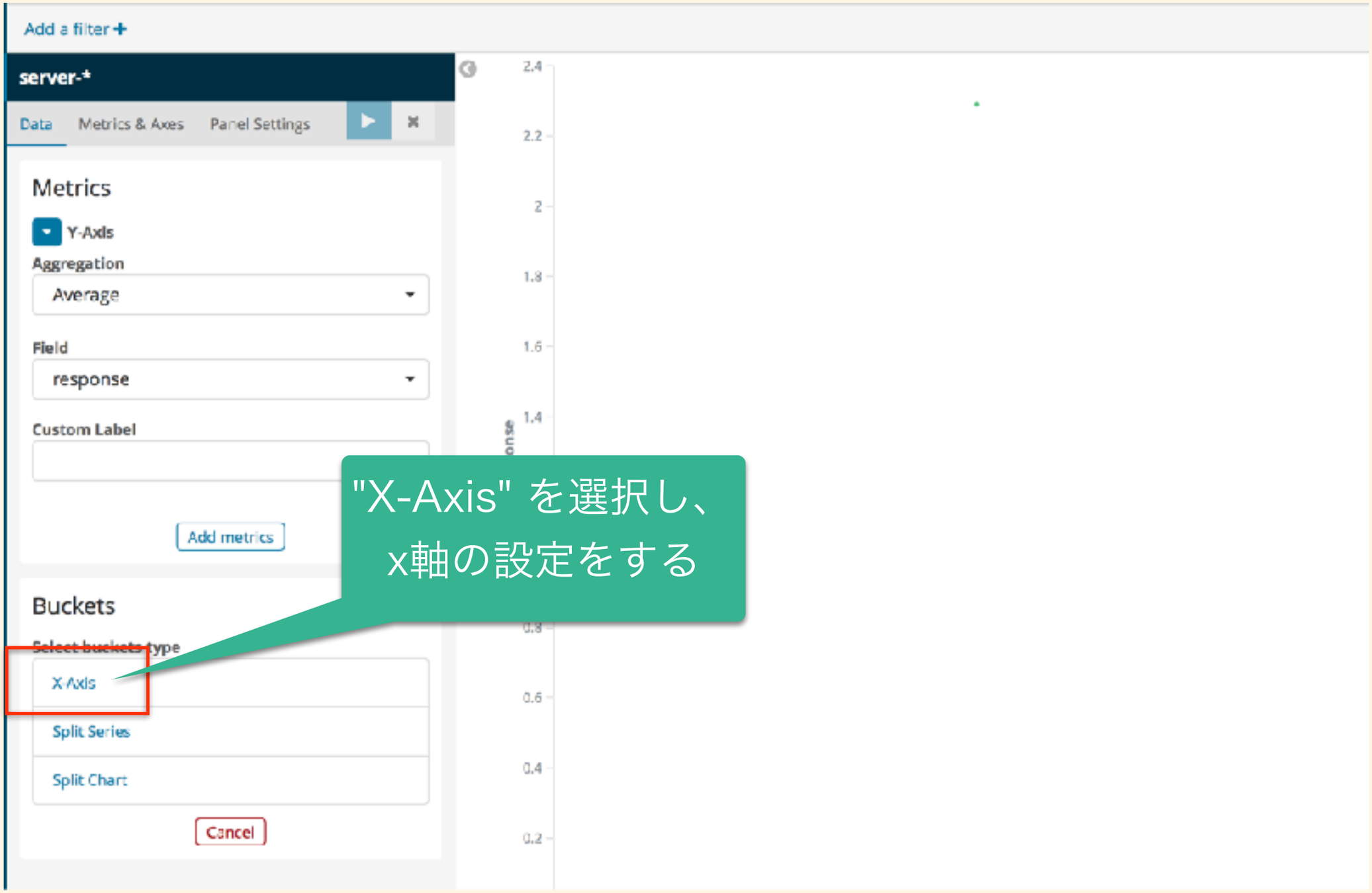
Visualizationの作成



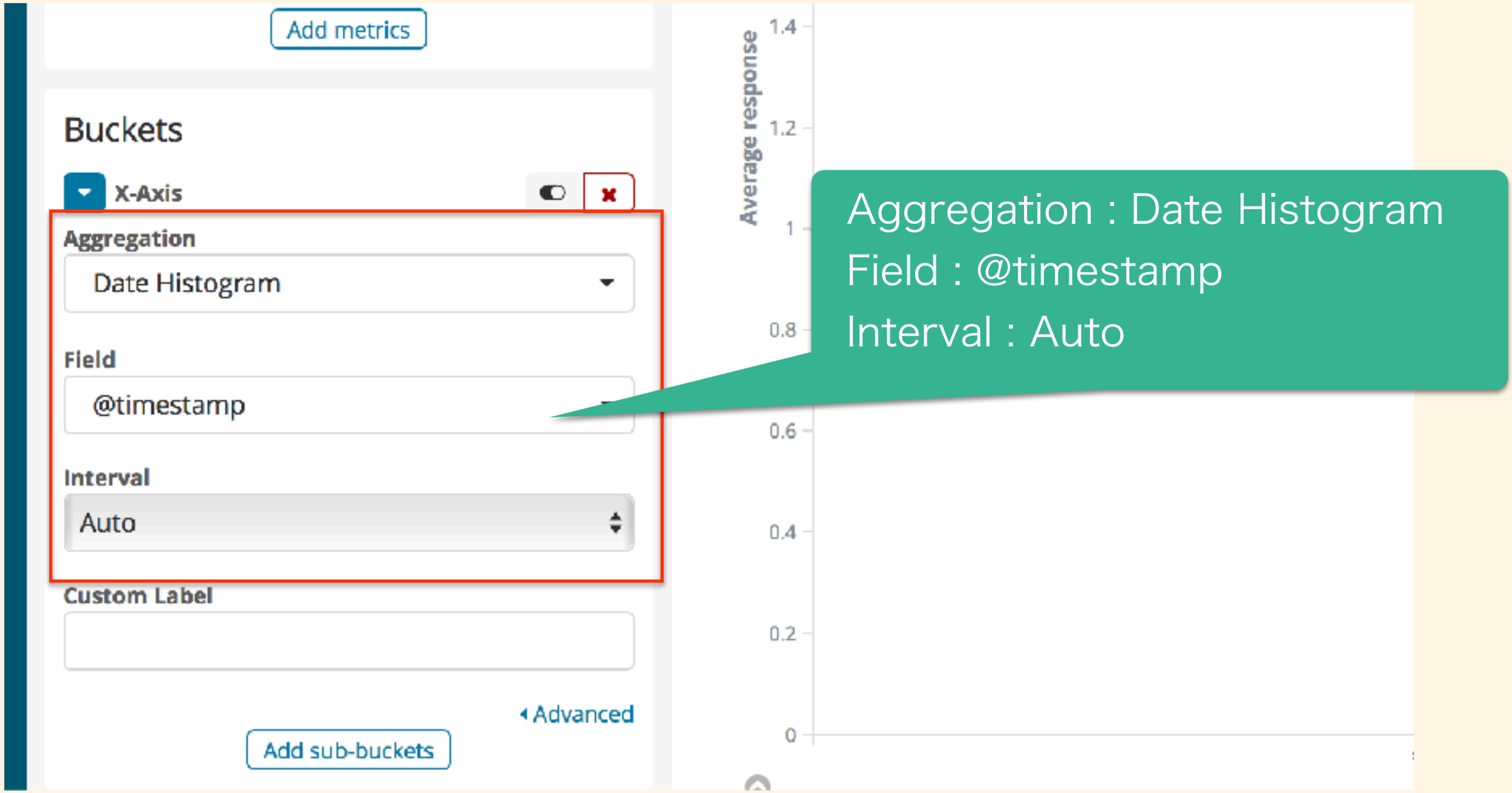
Visualizationの作成



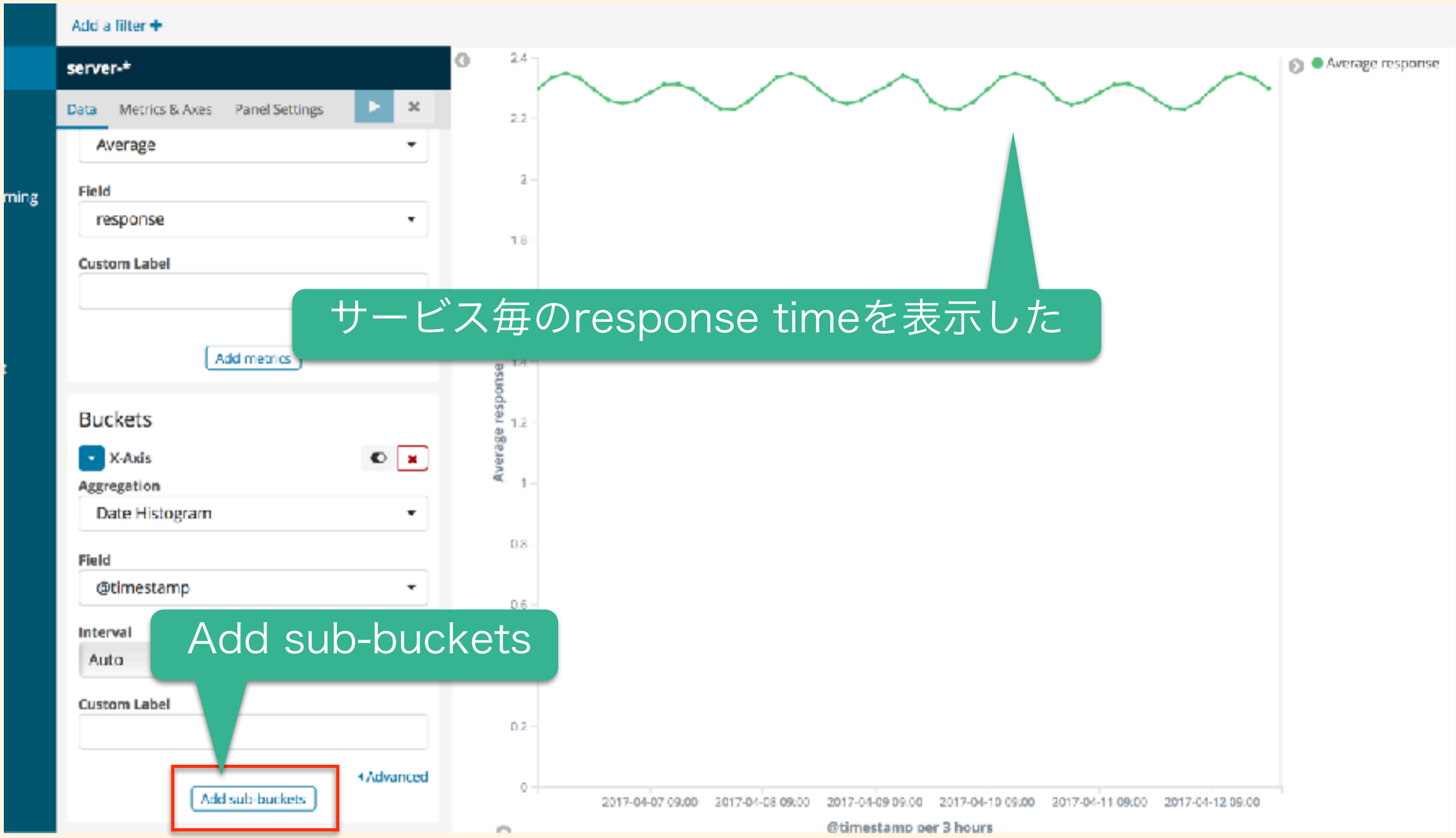
Visualizationの作成



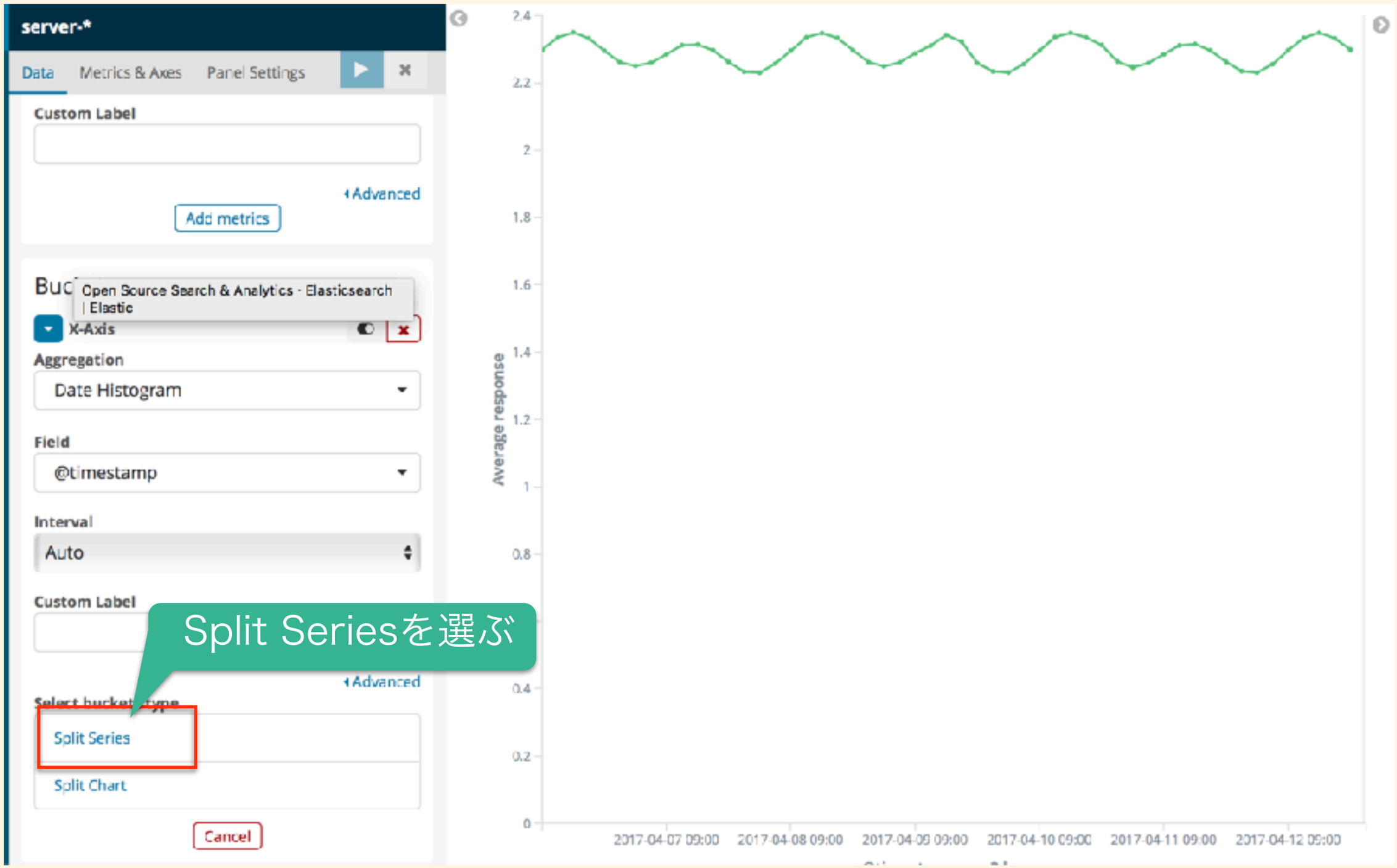
Visualizationの作成



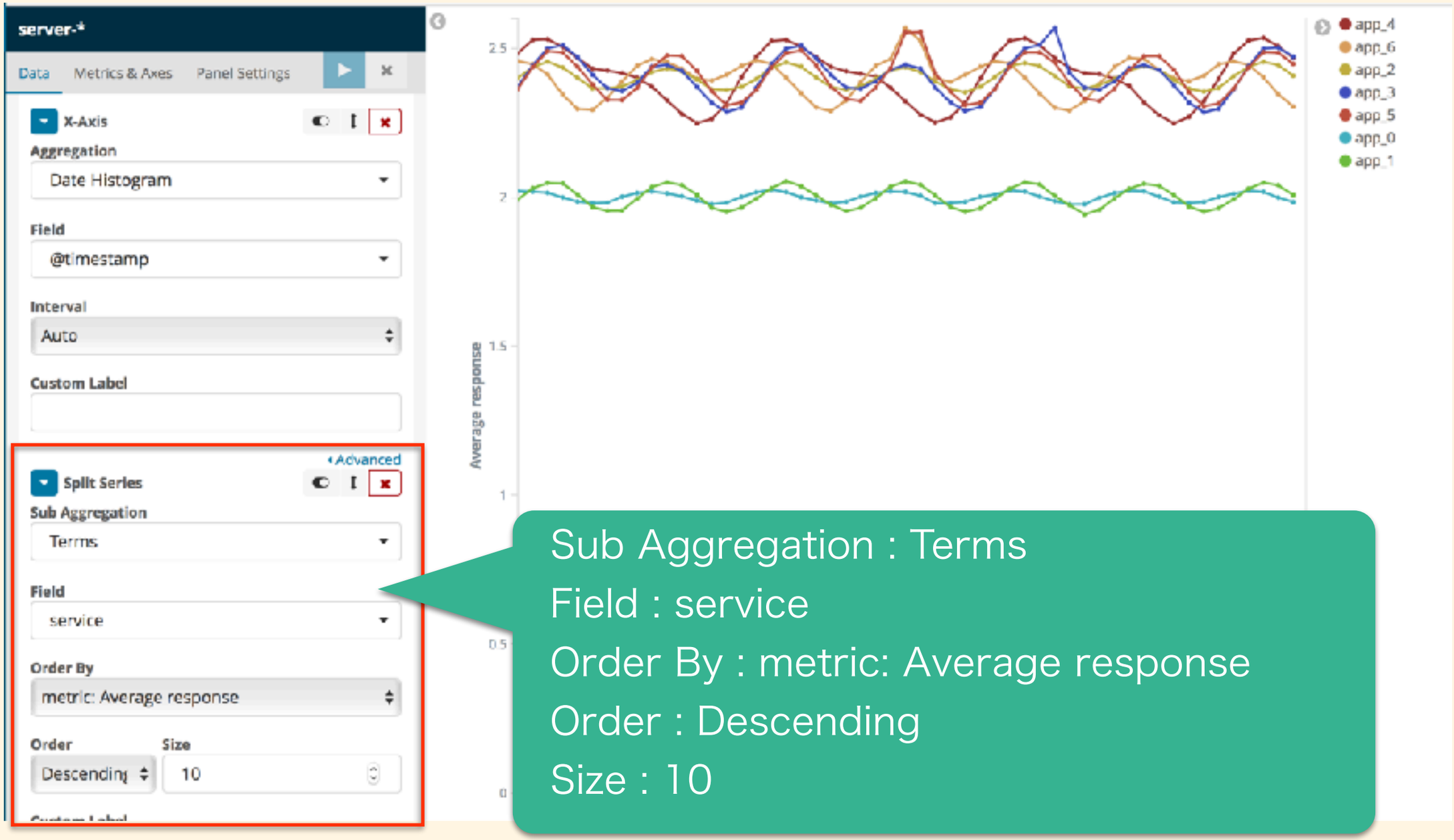
Visualizationの作成



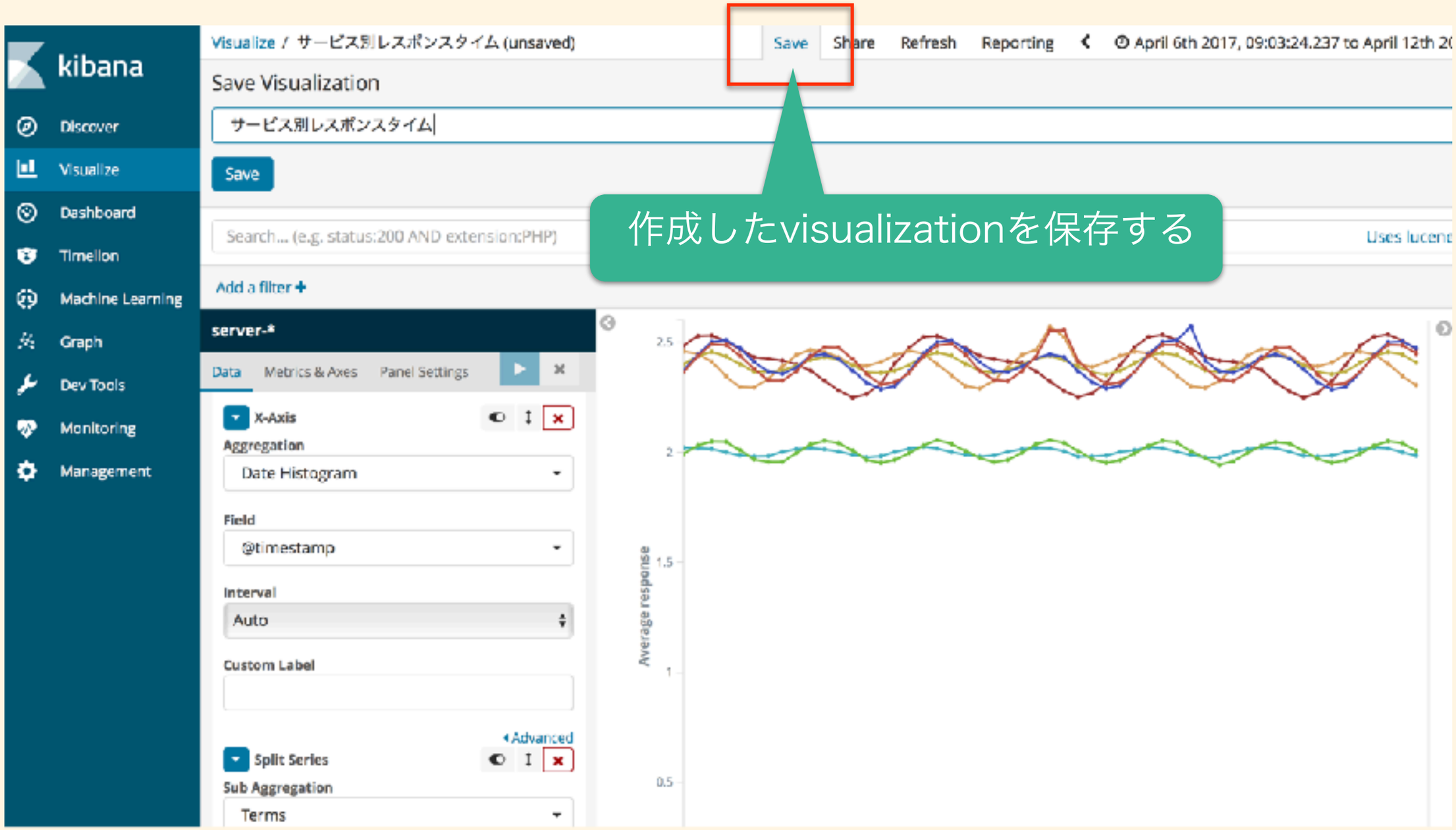
Visualizationの作成



Visualizationの作成

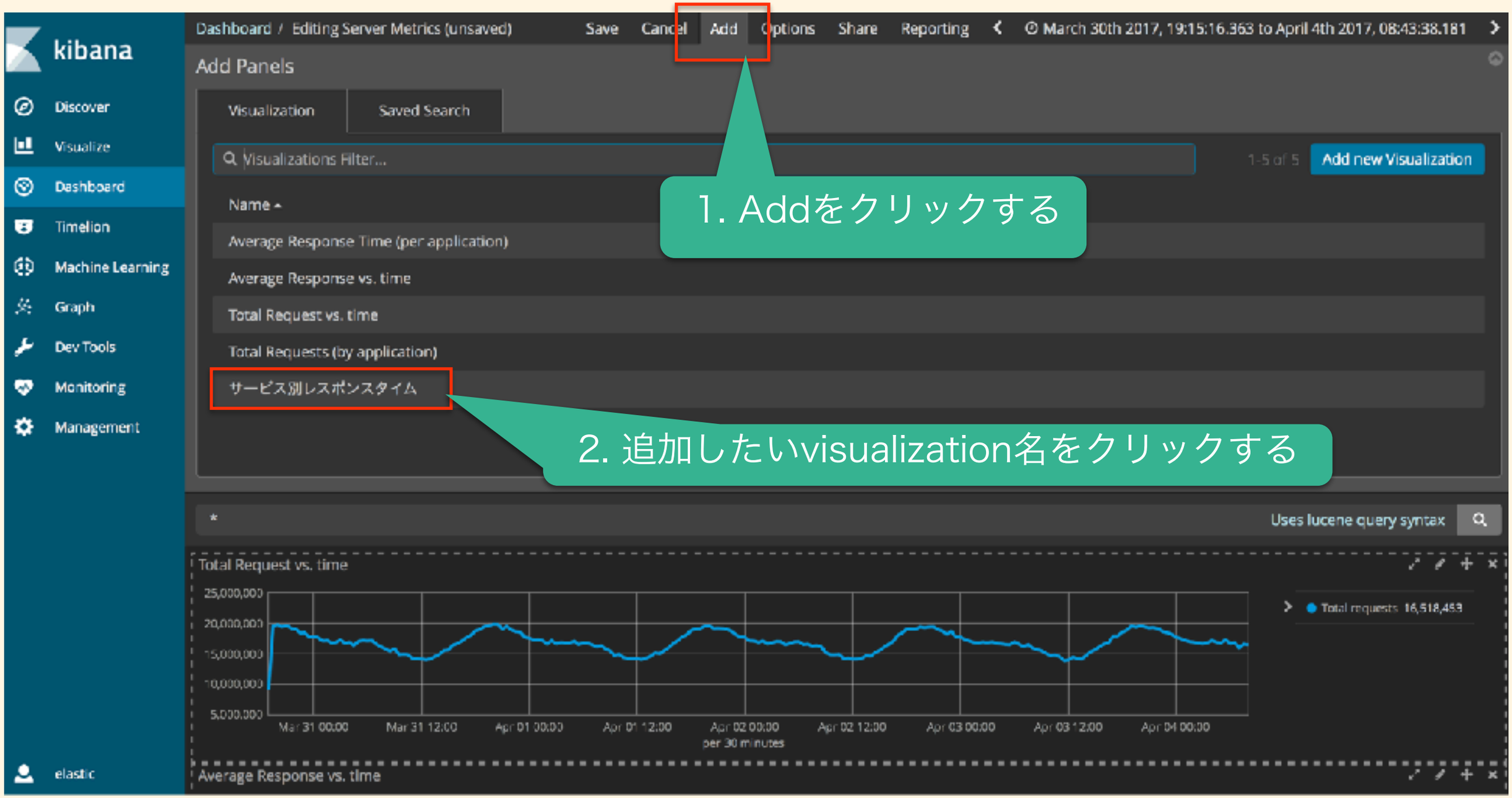


Visualizationの作成

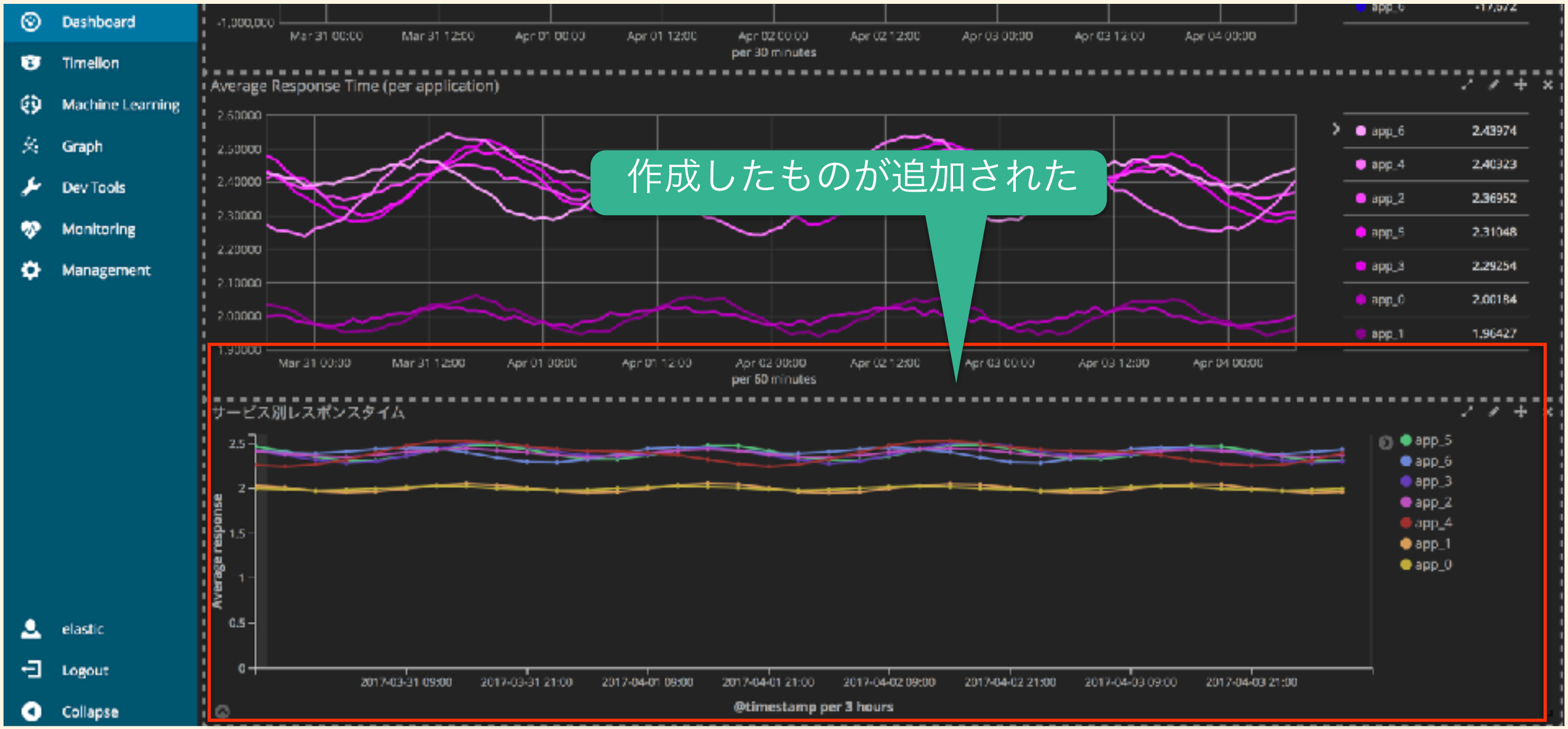


Dashboardの作成

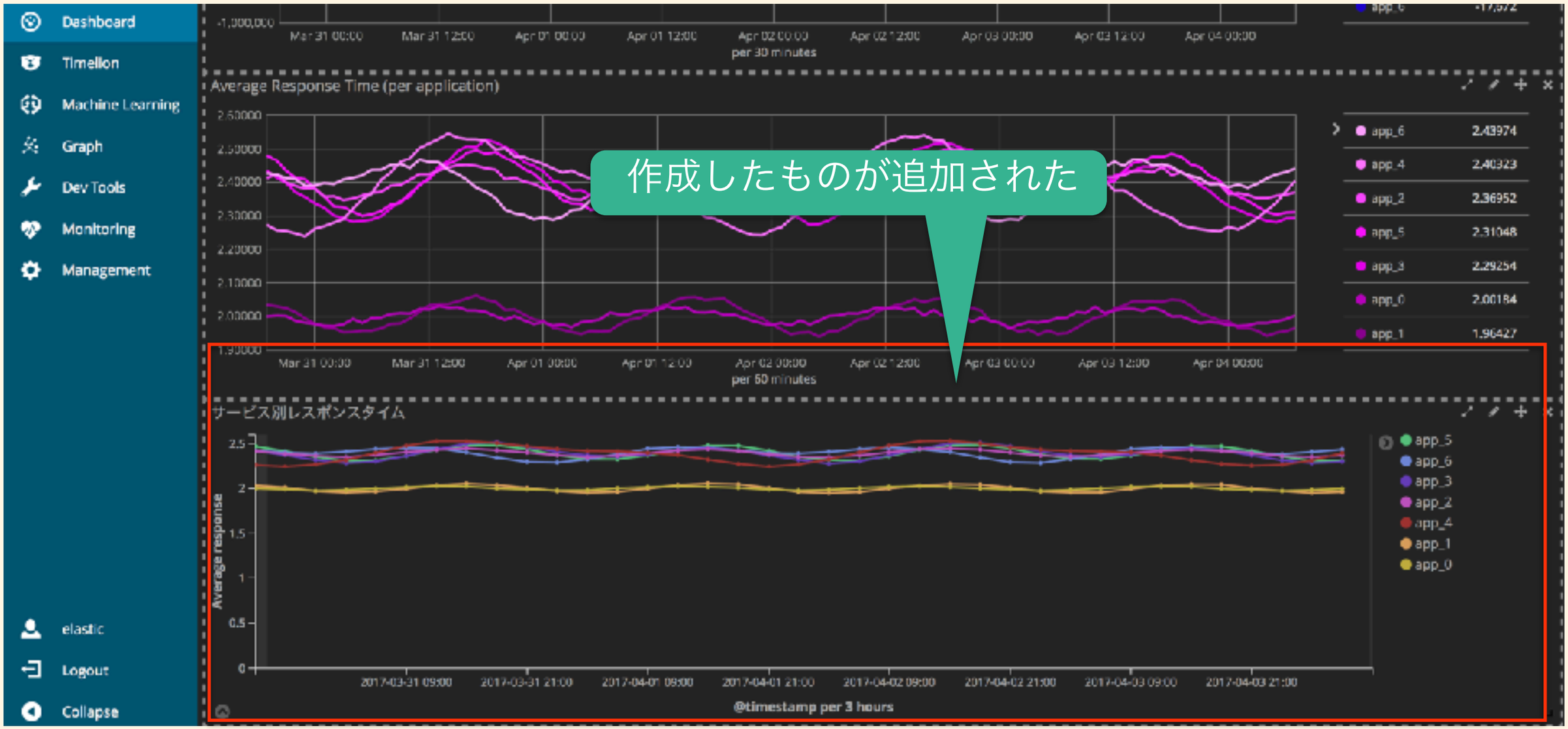
Dashboardの作成



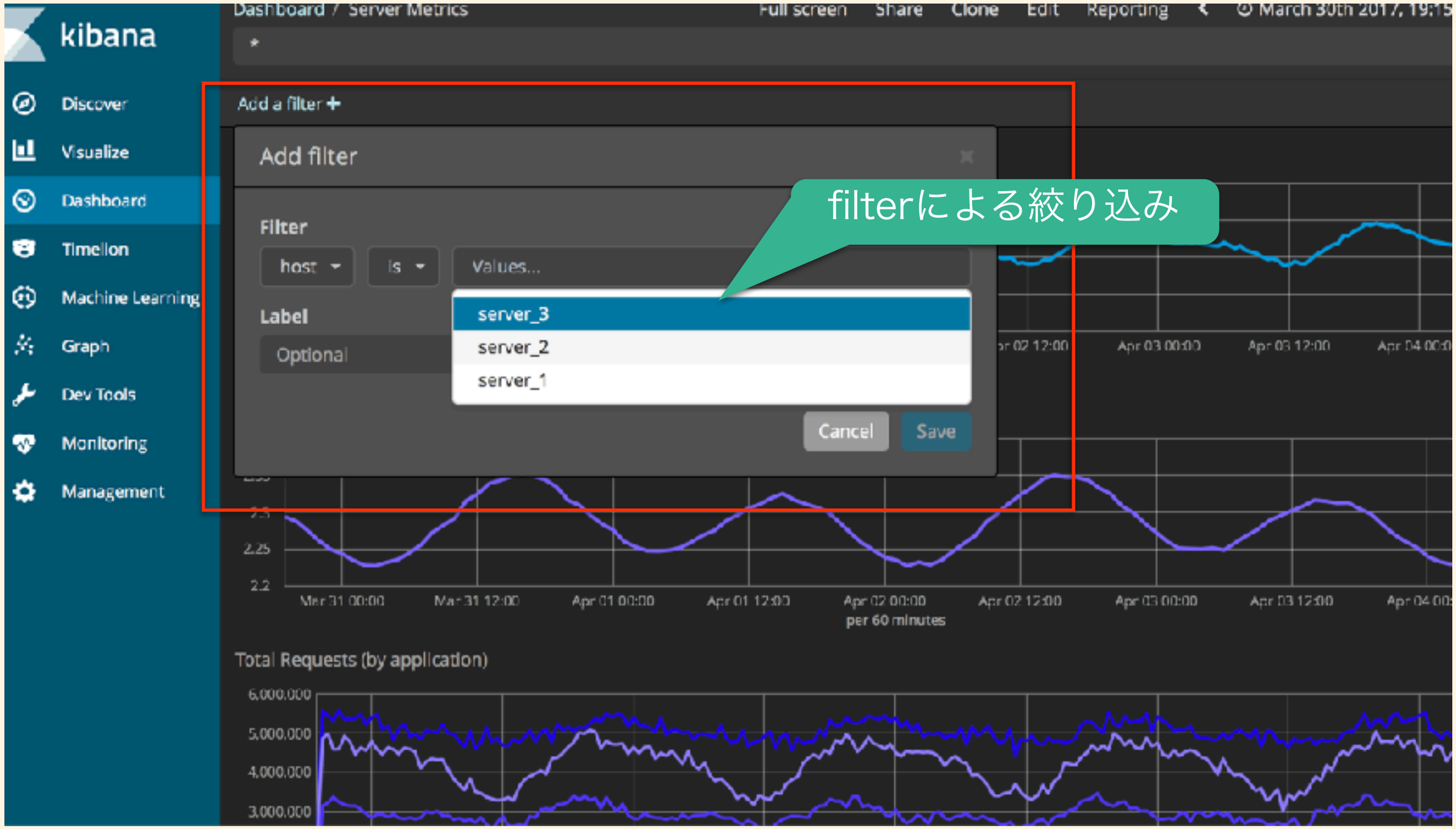
Dashboardの作成



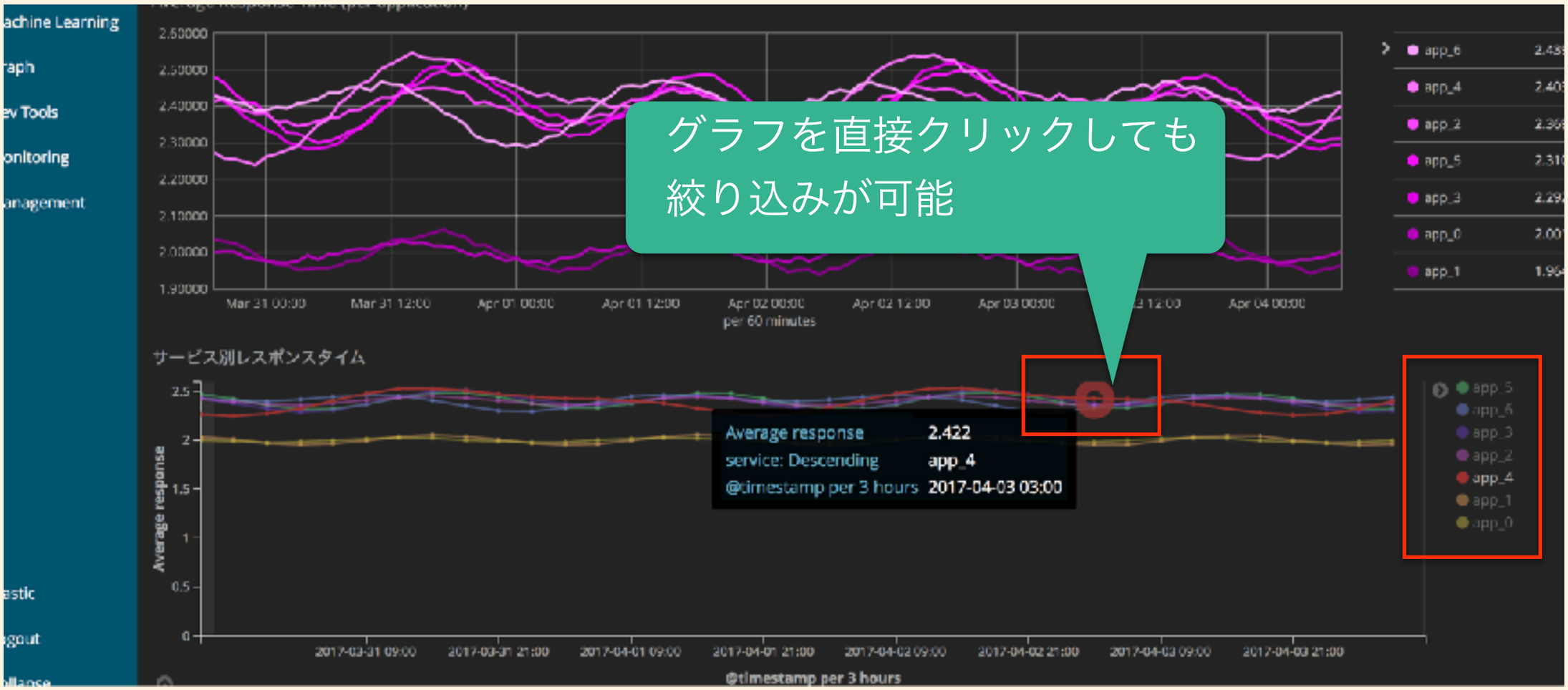
Dashboardの操作



Dashboardの操作



Dashboardの操作



Watcher UI

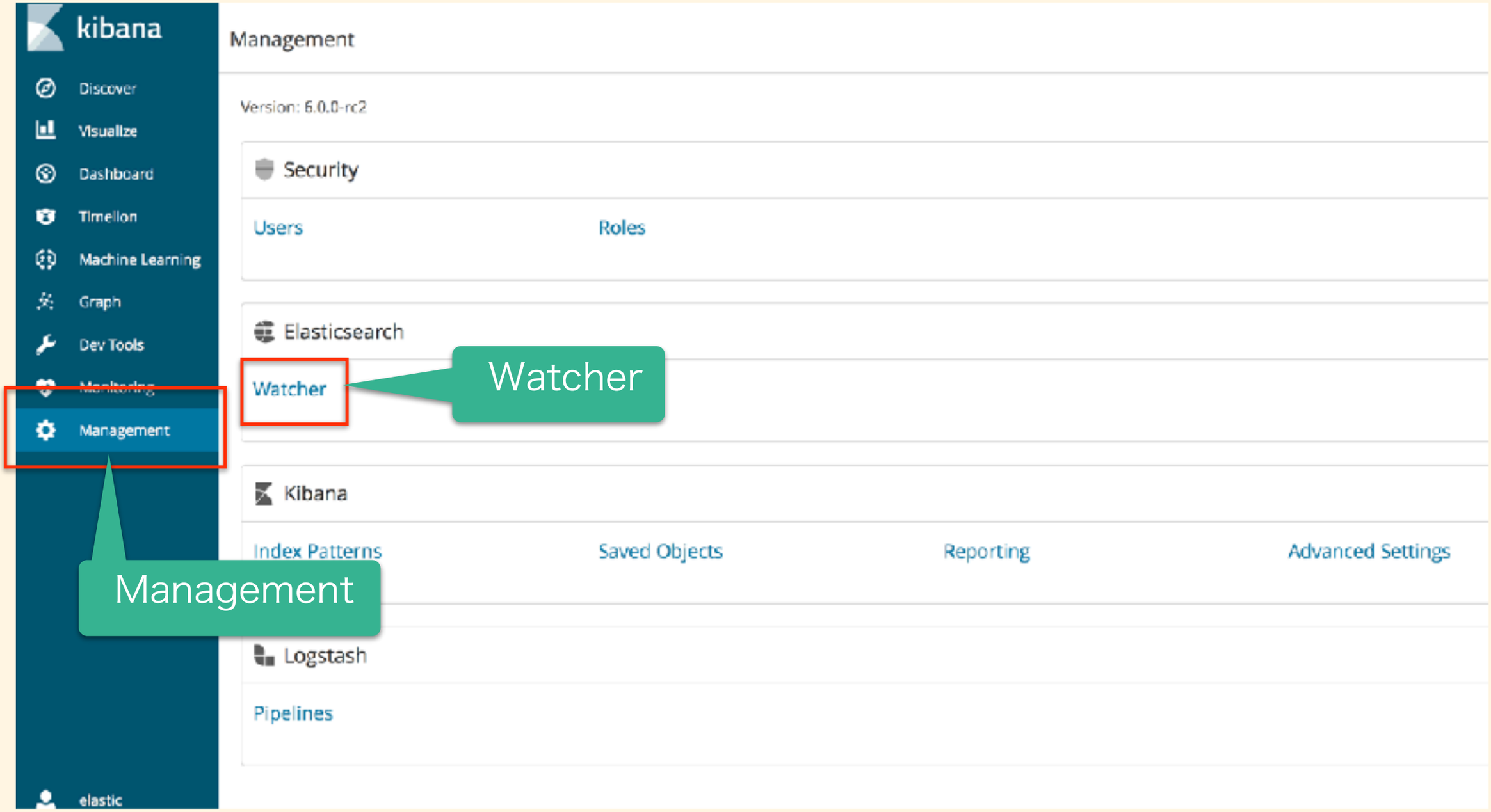
- Watcherとは
  - 特定の条件を満たした時に、
    メール・slackなどで通知をしたり、
    indexを作成することができる監視機能

例） CPU使用率が一定値を超えたらメール通知する

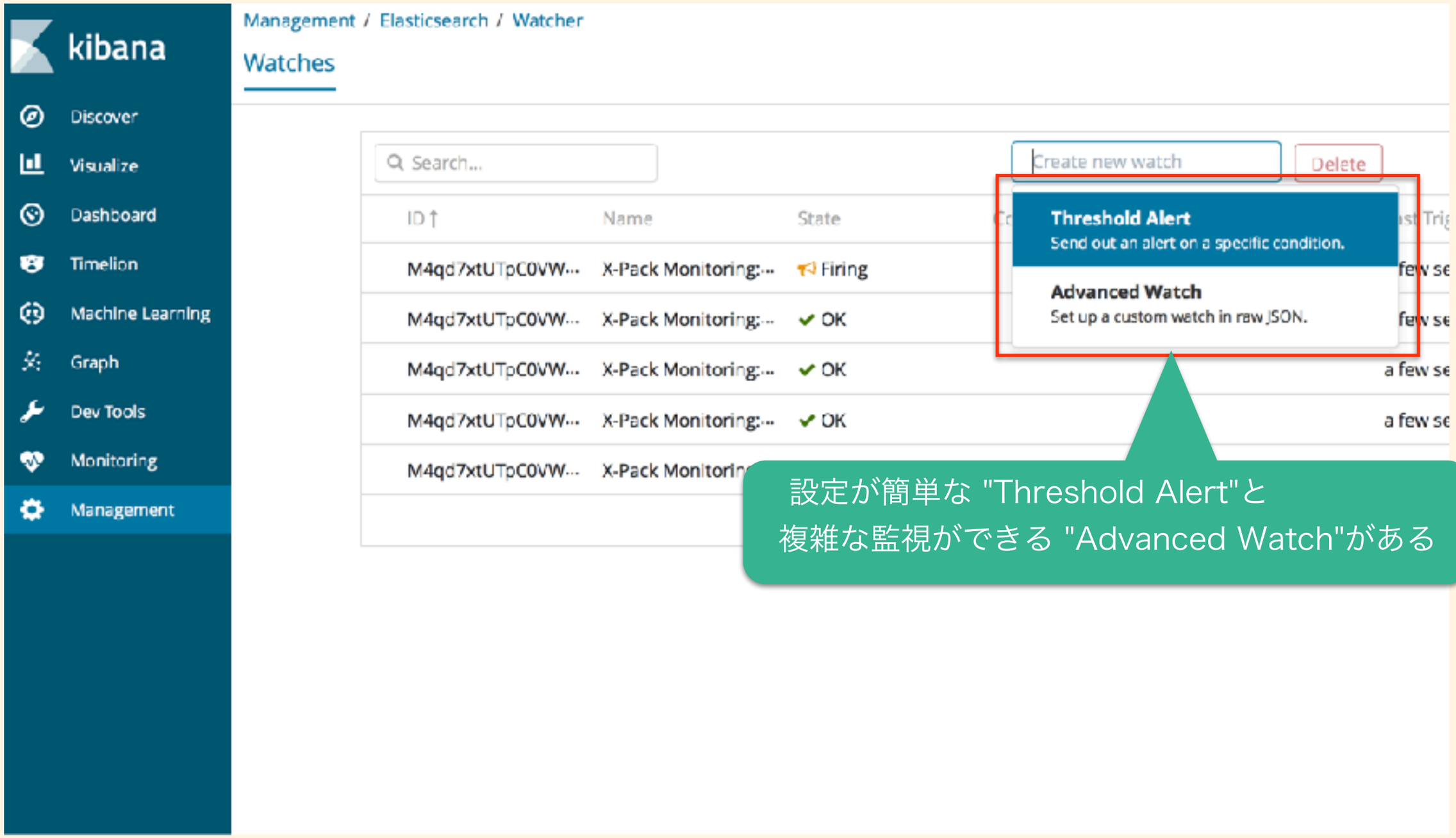
レスポンスタイムが閾値を超えたデータのみ別indexに登録する

アクセス数が急激に増加したらslackで通知する

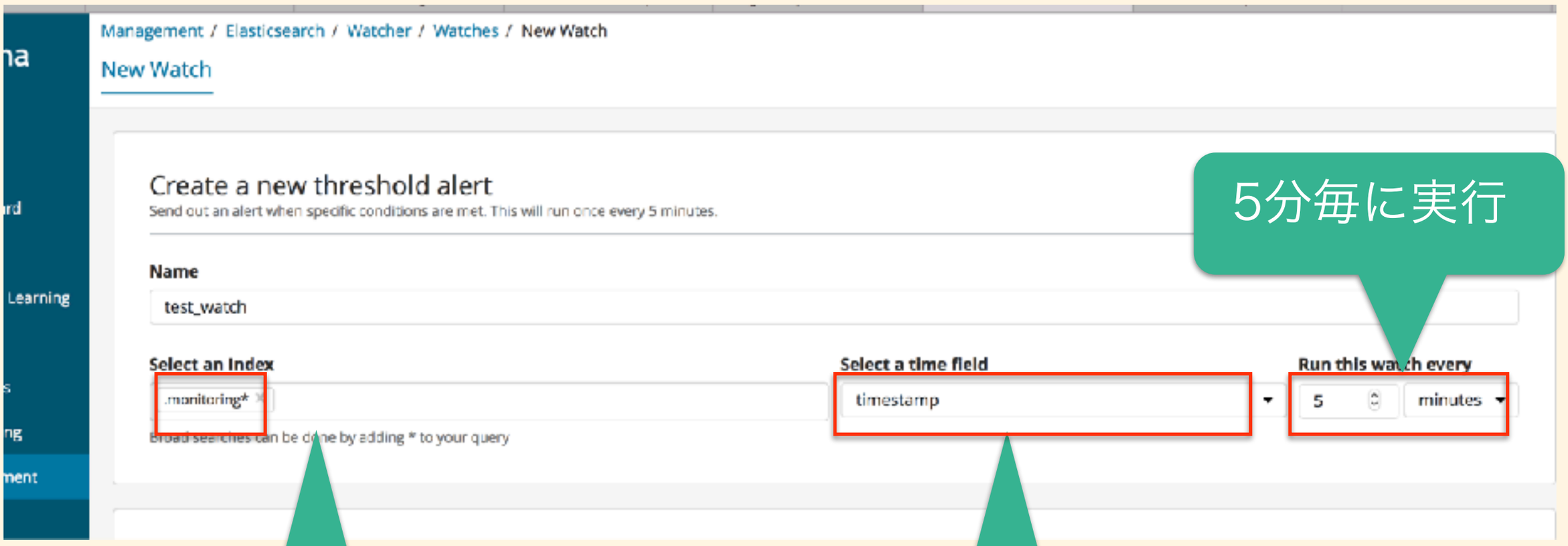
Watcher UI


kibana
Discover
Visualize
Dashboard
Timelion
Machine Learning
Graph
Dev Tools
Management
elastic
Management
Version: 6.0.0-rc2
Security
Users
Roles
Elasticsearch
Watcher
Watcher
Kibana
Index Patterns
Saved Objects
Reporting
Advanced Settings
Logstash
Pipelines
Management

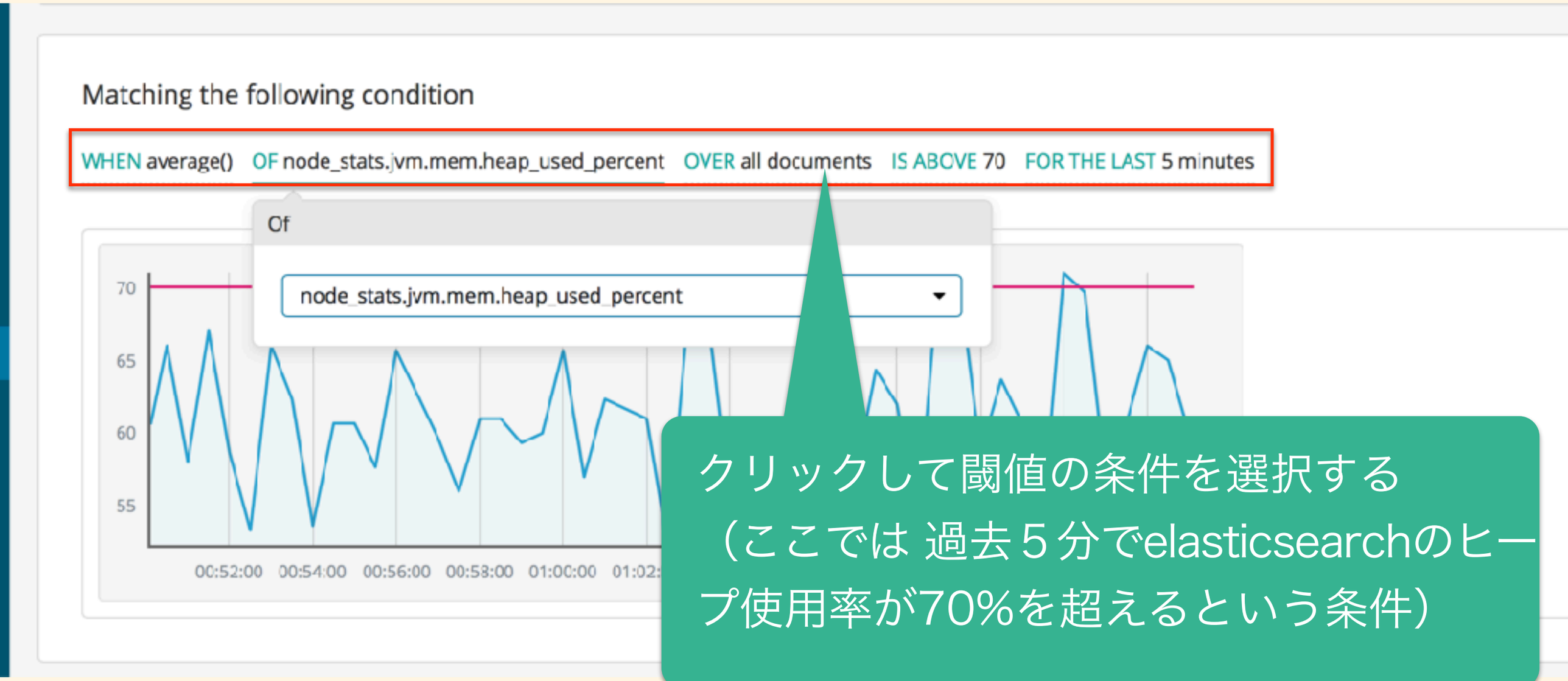
Watcher UI


kibana
Discover
Visualize
Dashboard
Timelion
Machine Learning
Graph
Dev Tools
Monitoring
Management
Management / Elasticsearch / Watcher
Watches
Search...
Create new watch
Delete
ID
Name
State
M4qd7xtUTpC0vW...
X-Pack Monitoring:...
Firing
OK
Threshold Alert
Send out an alert on a specific condition.
Advanced Watch
Set up a custom watch in raw JSON.
設定が簡単な "Threshold Alert"と
複雑な監視ができる "Advanced Watch"がある

Watcher UI


Management / Elasticsearch / Watcher / Watches / New Watch
New Watch
Create a new threshold alert
Send out an alert when specific conditions are met. This will run once every 5 minutes.
Name
test_watch
Select an Index
.monitoring*
Broad searches can be done by adding * to your query
Select a time field
timestamp
Run this watch every
5
minutes
5分毎に実行
index : .monitoring*
@timestamp

Watcher UI


Matching the following condition
WHEN average() OF node_stats.jvm.mem.heap_used_percent OVER all documents IS ABOVE 70 FOR THE LAST 5 minutes
Of
node_stats.jvm.mem.heap_used_percent
70
65
60
55
00:52:00
00:54:00
00:56:00
00:58:00
01:00:00
クリックして閾値の条件を選択する
（ここでは 過去５分でelasticsearchのヒープ使用率が70%を超えるという条件）

Watcher UI

条件を満たした時に実行する
「アクション」を設定する

Watcher UI


Monitoring
Management
elastic
Logout
Will perform 1 action once met
Add new action
Logging
Log text
これはWatcherのテスト
Rem Logging Action
Log a sample message now
Save
2. 保存する前に動作を確認できる
1. loggingアクションを選び
ログに出力したいメッセージを入力

Watcher UI

```
d away from this node
[2017-11-18T01:22:05,763][WARN ][o.e.c.r.a.DiskThresholdMonitor] [ZV1XgHZ] high disk watermark [90%] exceeded on [ZV1XgHZCR
5rOEkw][ZV1XgHZ][/Users/SHIN/jjug_2017_fall/work/elasticsearch-6.0.0-rc2/data/nodes/0] free: 14.8gb[6.3%], shards will be r
d away from this node
[2017-11-18T01:22:05,763][INFO ][o.e.c.r.a.DiskThresholdMonitor] [ZV1XgHZ] rerouting shards: [high disk watermark exceeded
or more nodes]
[2017-11-18T01:22:35,865][WARN ][o.e.c.r.a.DiskThresholdMonitor] [ZV1XgHZ] high disk watermark [90%] exceeded on [ZV1XgHZCR
5rOEkw][ZV1XgHZ][/Users/SHIN/jjug_2017_fall/work/elasticsearch-6.0.0-rc2/data/nodes/0] free: 14.8gb[6.3%], shards will be r
d away from this node
[2017-11-18T01:22:59,337][INFO ][o.e.x.w.a.l.ExecutableLoggingAction] [ZV1XgHZ] hogehoge
[2017-11-18T01:23:03,111][INFO ][o.e.x.w.a.l.ExecutableLoggingAction] [ZV1XgHZ] hogehoge
[2017-11-18T01:23:05,867][WARN ][o.e.c.r.a.DiskThresholdMonitor] [ZV1XgHZ] high disk watermark [90%] exceeded on [ZV1XgHZCR
5rOEkw][ZV1XgHZ][/Users/SHIN/jjug_2017_fall/work/elasticsearch-6.0.0-rc2/data/nodes/0] free: 14.8gb[6.3%], shards will be r
d away from this node
[2017-11-18T01:23:05,867][INFO ][o.e.c.r.a.DiskThresholdMonitor] [ZV1XgHZ] rerouting shards: [high disk watermark exceeded
or more nodes]
[2017-11-18T01:23:25,313][INFO ][o.e.x.w.a.l.ExecutableLoggingAction] [ZV1XgHZ] これはWatcherのテスト
```

ログに出力されている

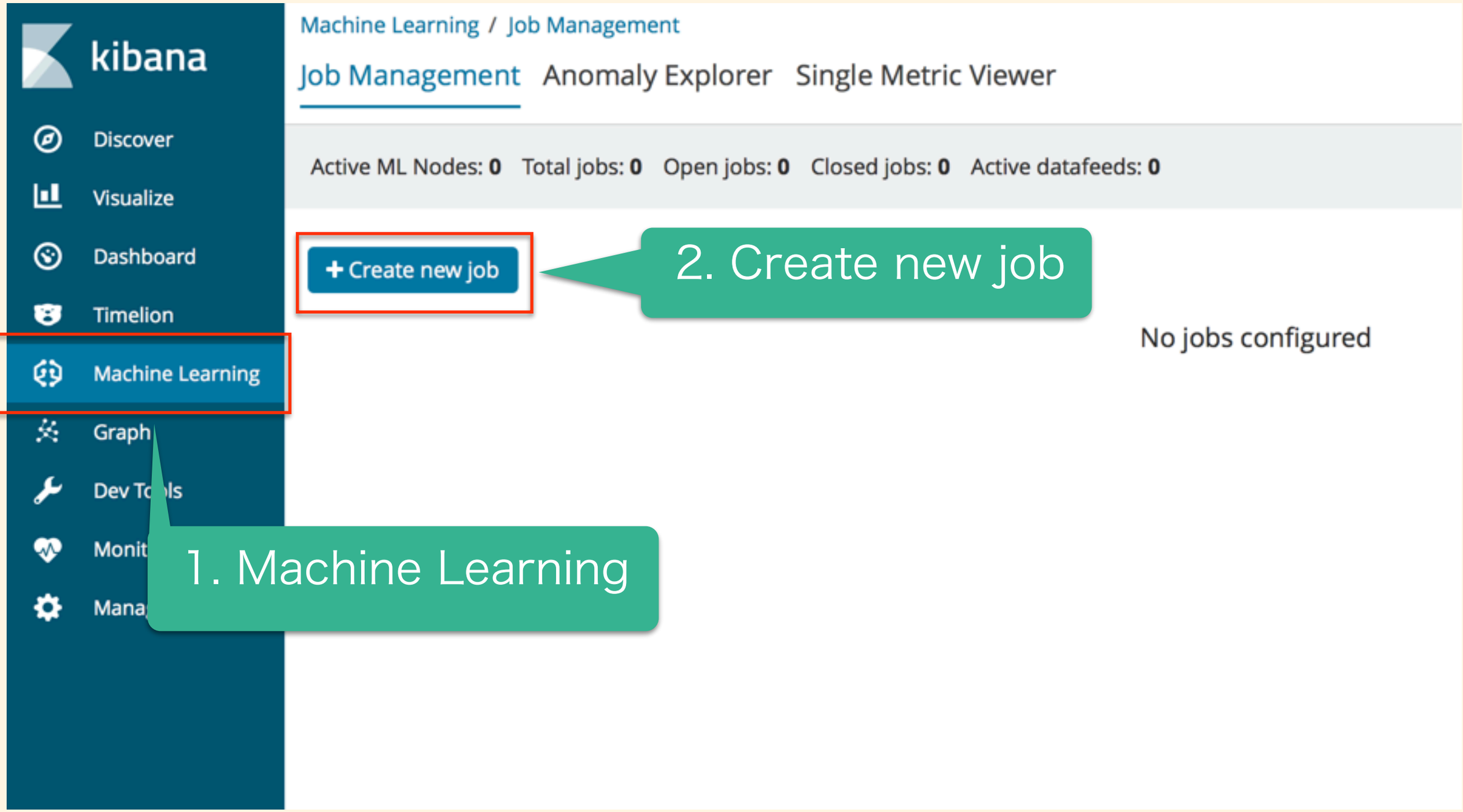
コンテンツ

- 概要説明
- データの投入
- ダッシュボードの作成
- Machine Learning ←
- まとめ

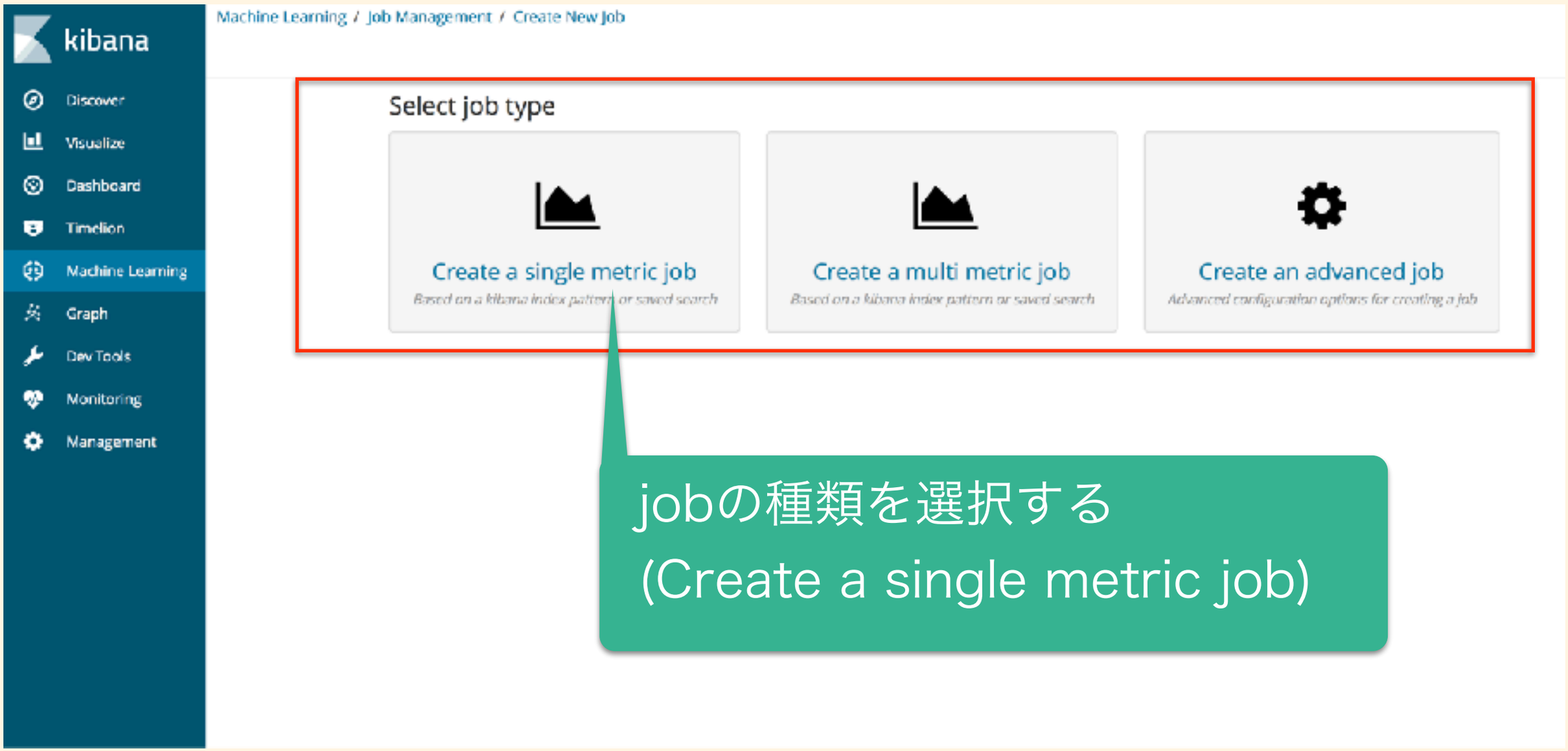
Machine Learning

- Machine Learningとは
    - X-Packの機能の１つ
    - 時系列データの傾向を学習し、異常を検知する。(教師無し学習)

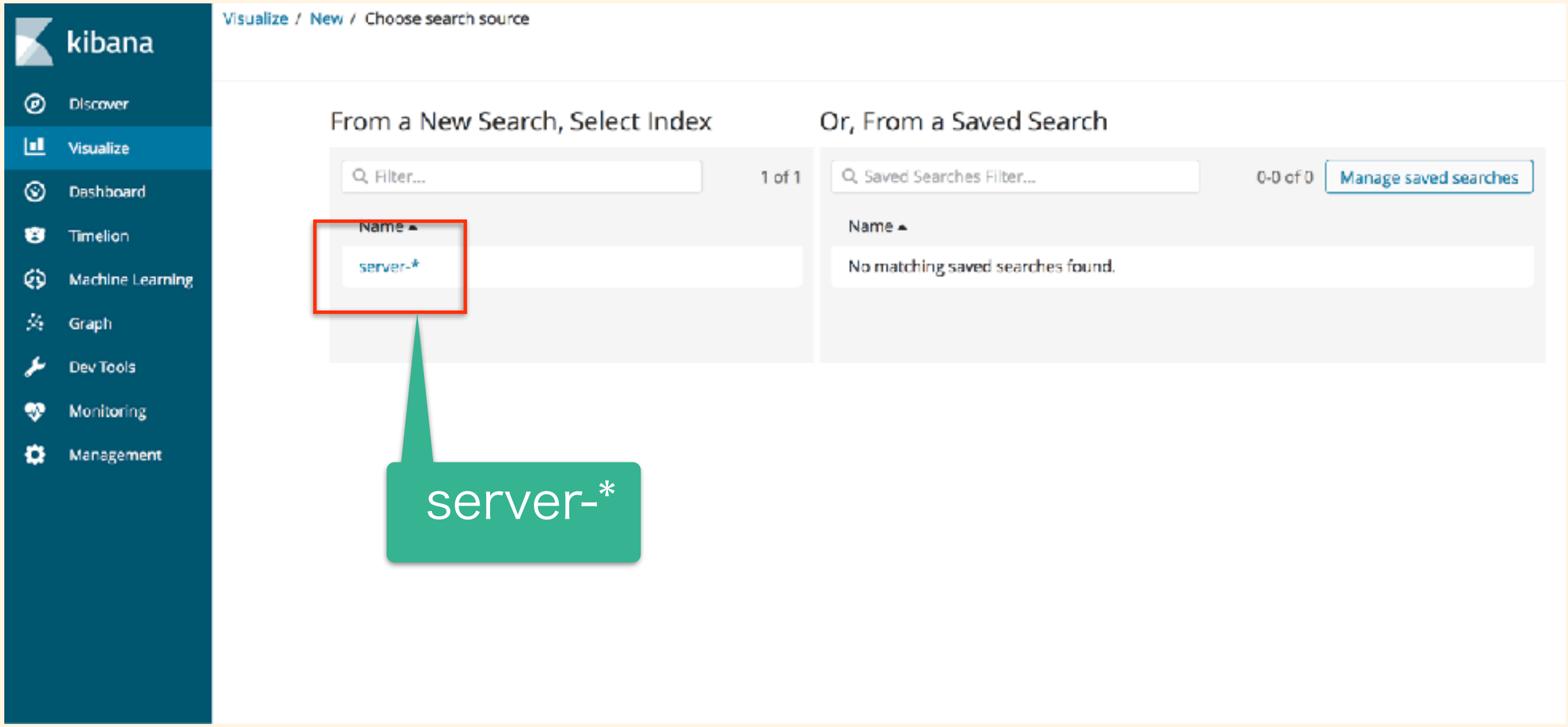
Machine Learning



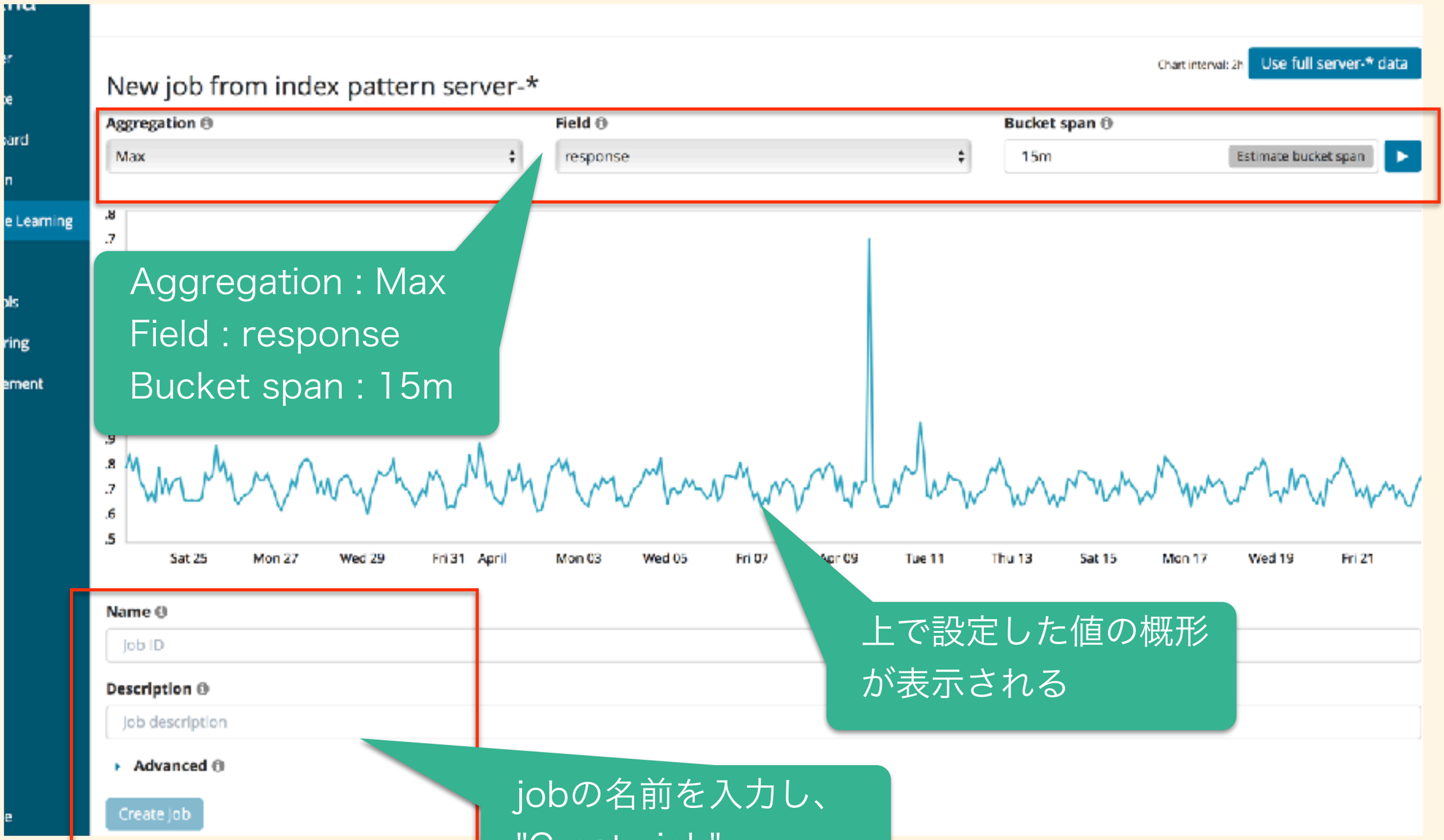
Machine Learning



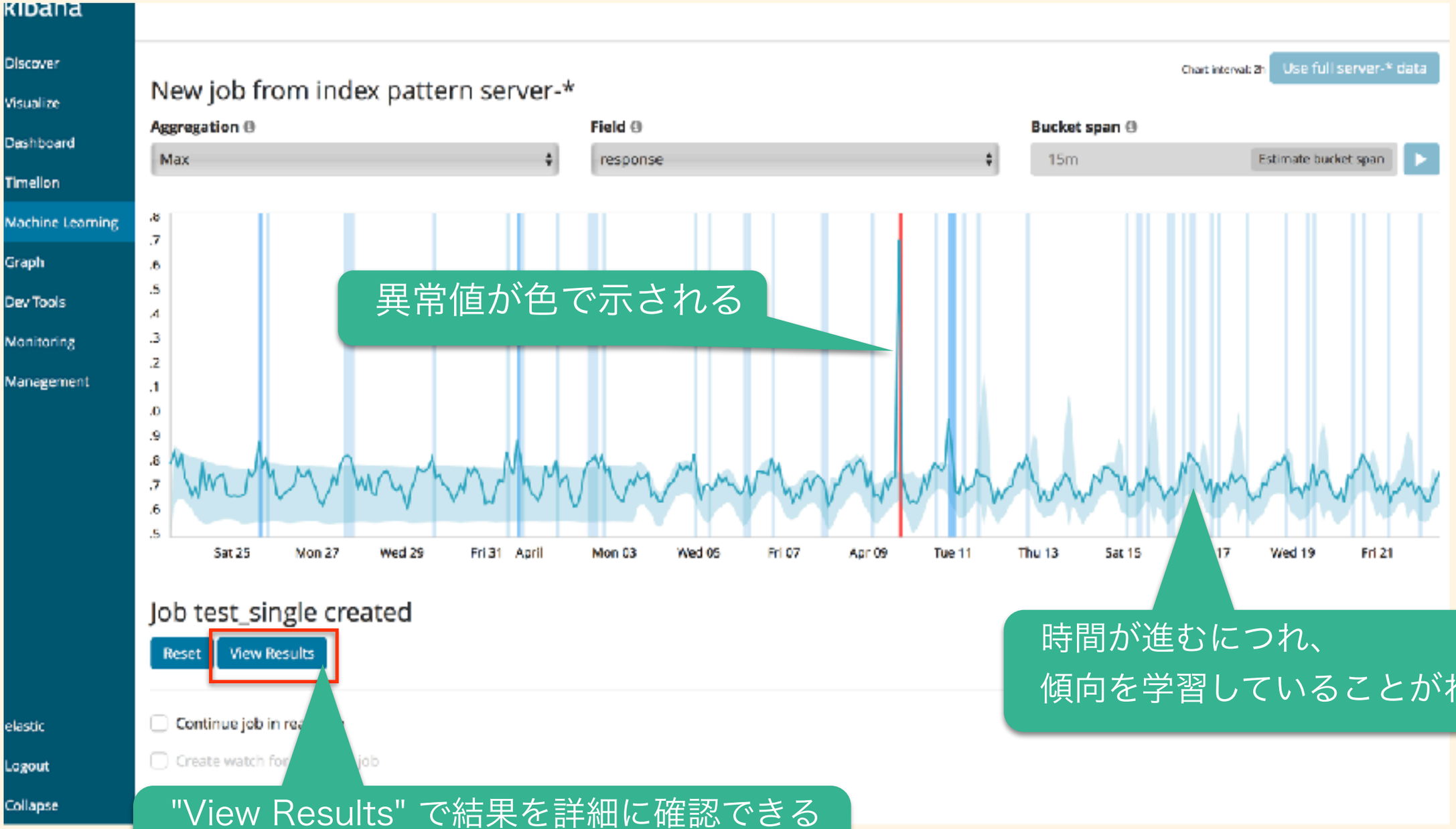
Machine Learning



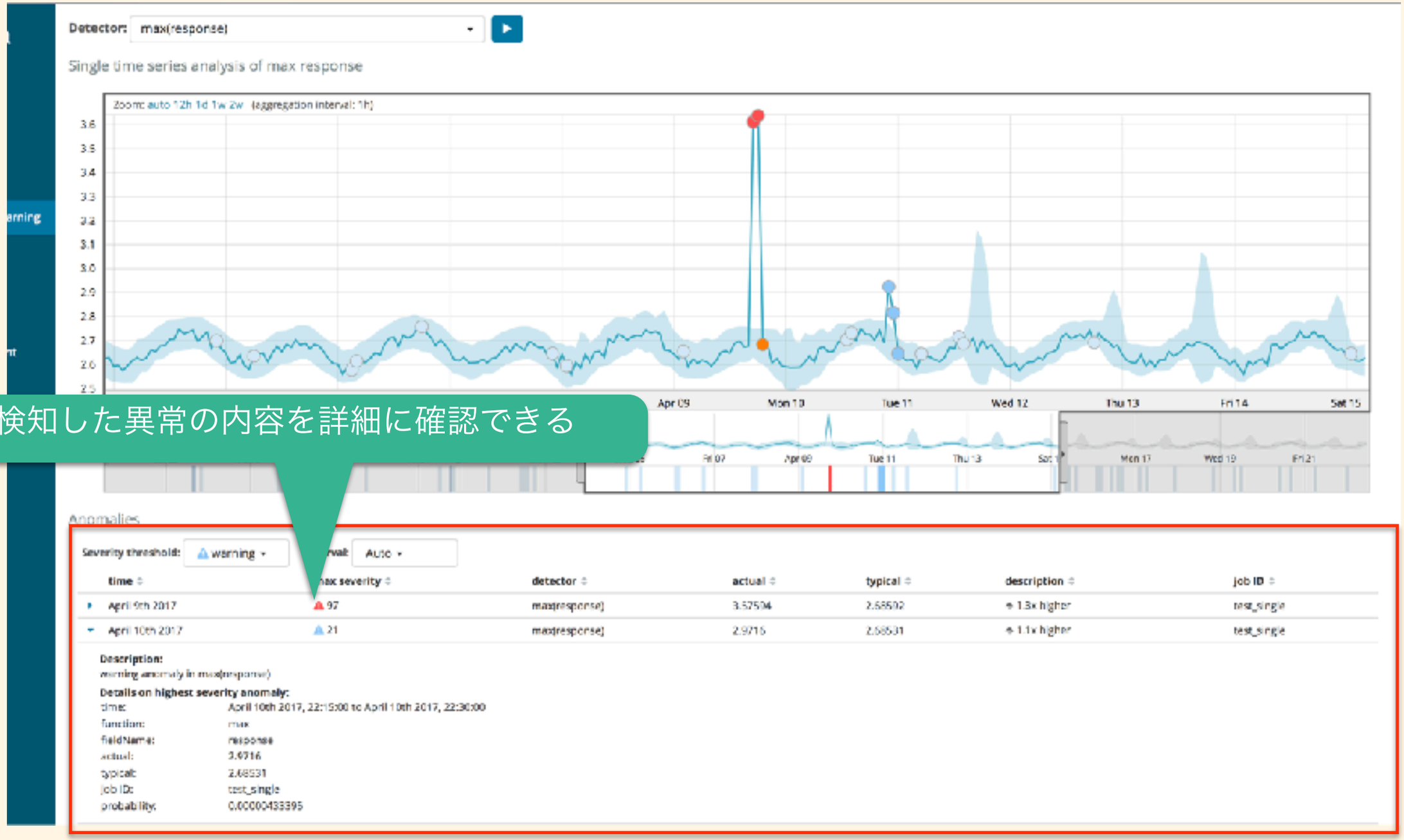
Machine Learning



Machine Learning



Machine Learning

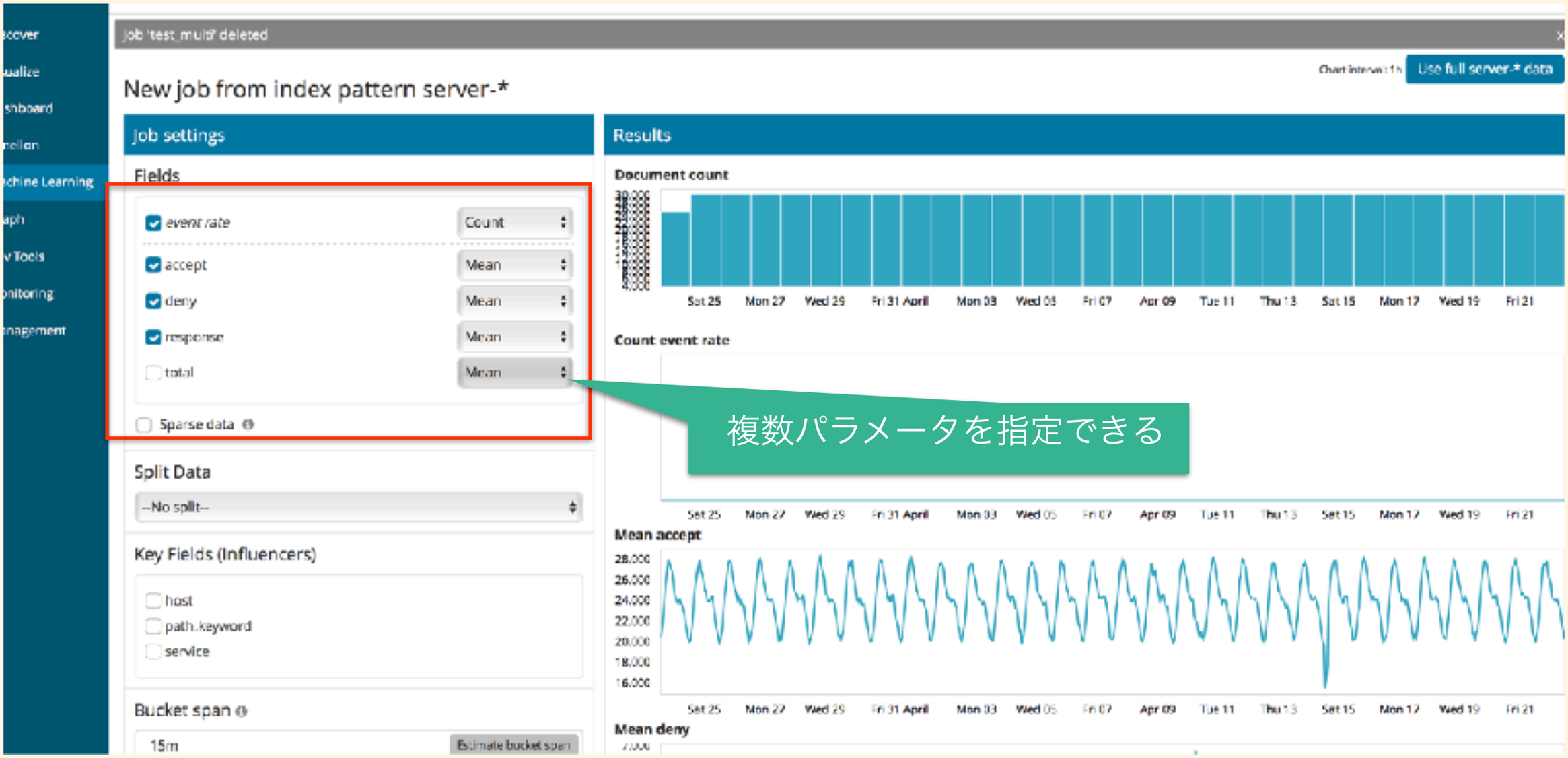


検知した異常の内容を詳細に確認できる

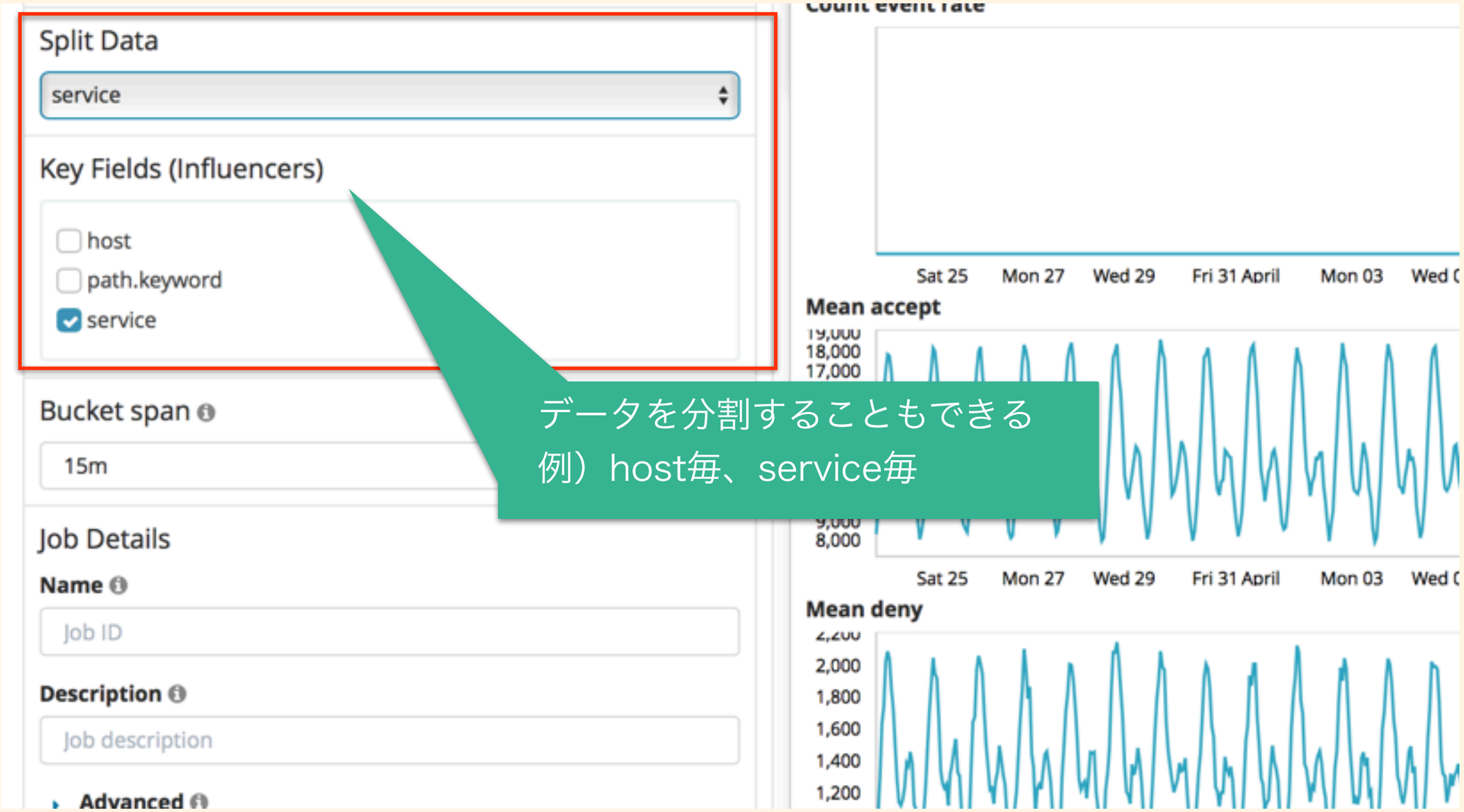
発展

Machine Learning

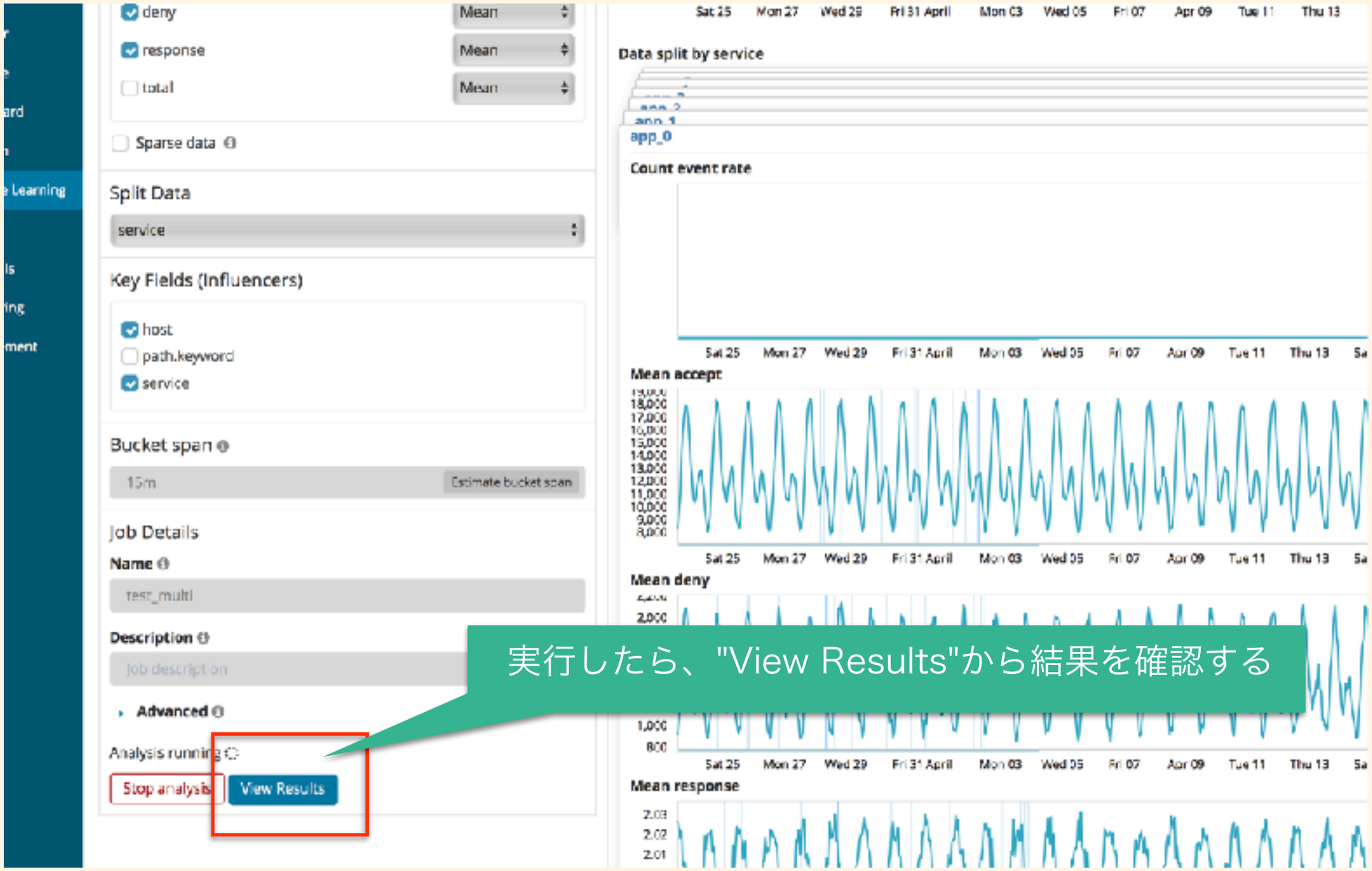
Machine Learning



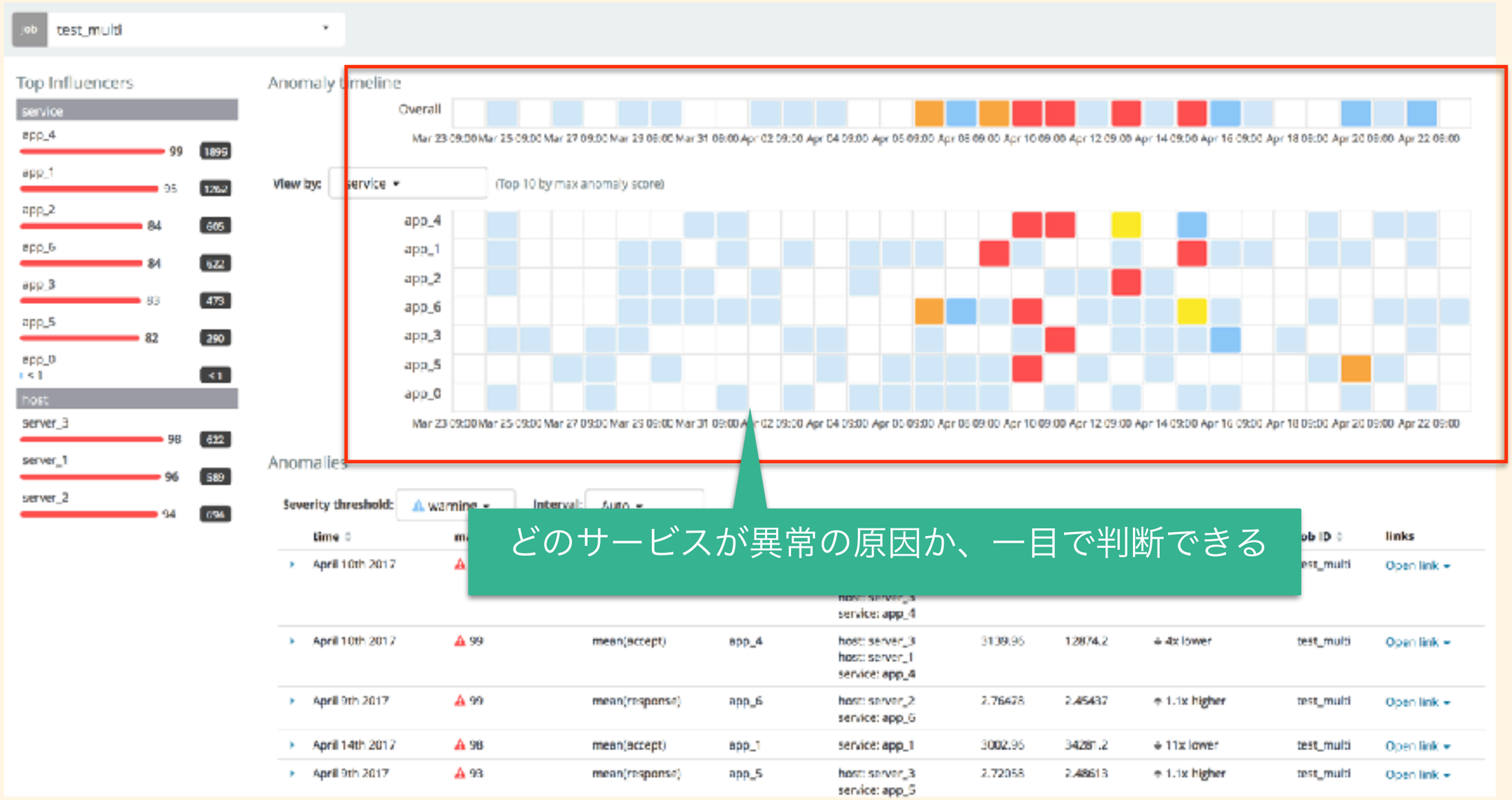
Machine Learning



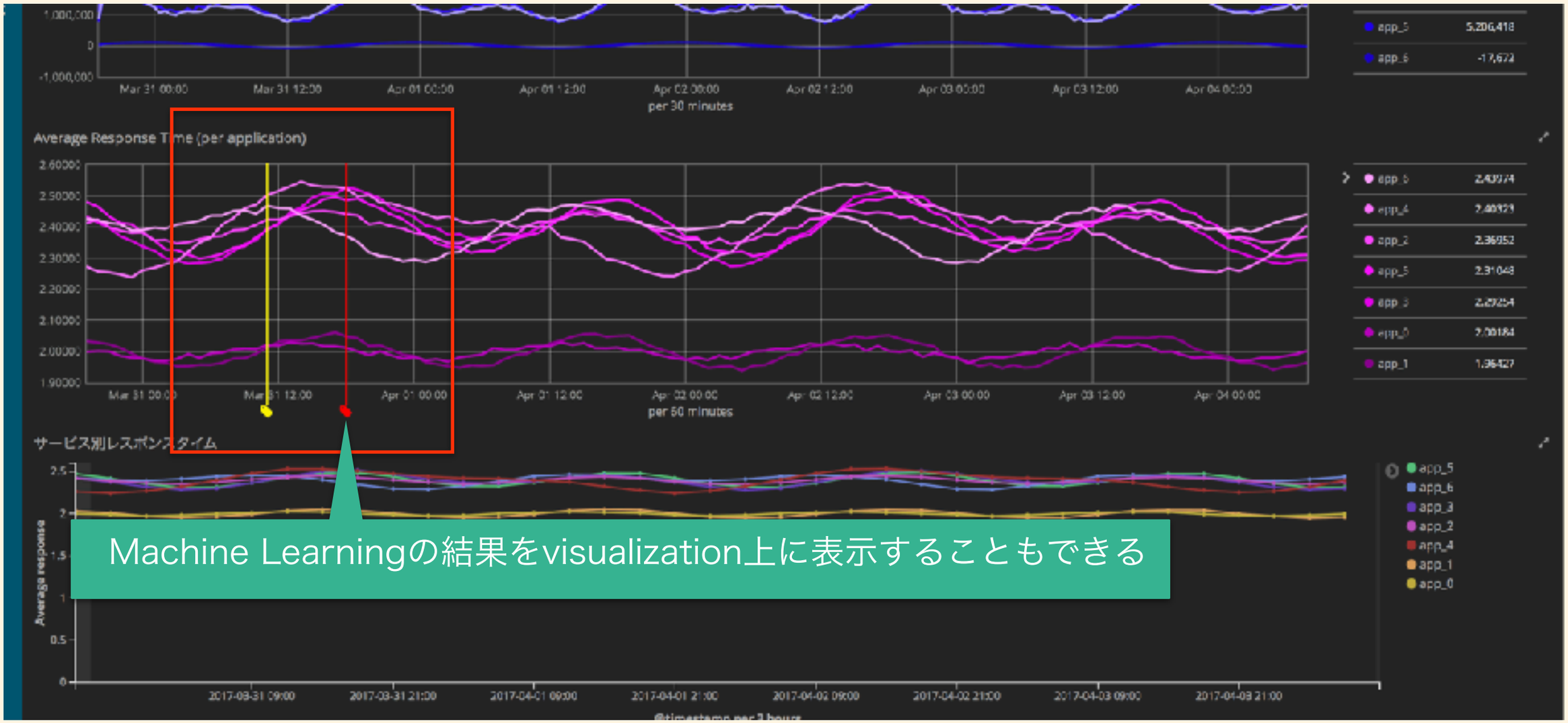
Machine Learning



Machine Learning



Machine Learning



余力のある方は試してみてください！

コンテンツ

- 概要説明
- データの投入
- ダッシュボードの作成
- Machine Learning
- まとめ　←

まとめ

- 5.x → 6.0.0で追加された機能に触れた
  - logstash pipeline visualizer
  - Machine Learning
  - Watcher UI
  - Visual Builder

- 今回紹介できなかった改善点も多くあります。
  どんどん使ってみてください。

以上です。

ありがとうございました。