

System Investigation

Elasticsearch



Nayantara Mohan
Shubha Changappa Palachanda

Introduction to the Elastic Stack

The Elastic Stack

Reliably and securely takes data from a given source, in multiple formats and then searches, analyzes and visualizes it in real time.

Kibana
Explore, Visualize, Engage

Elasticsearch
Store, Search, Analyze

Integrations
Connect, Collect, Alert

Project: Dataset (Update)

- Join functions don't work on Elasticsearch like in RDBMS.
- We have to denormalise the dataset and flatten the file which can be done through logstash during the data ingestion phase.
- Denormalization increases the memory consumption which is a limitation given the cloud storage constraints.
- This increases the dataset size which is a limitation in the free / trial version of elastic cloud.
- Thus, to be able to use elastic cloud and investigate further on the search operations (strength of ES), we choose a smaller and a text richer dataset - Netflix from Kaggle.

Dataset Description

- Netflix Dataset from Kaggle

About this Dataset:

Netflix is one of the most popular media and video streaming platforms. They have over 8000 movies or tv shows available on their platform, as of mid-2021, they have over 200M Subscribers globally. This tabular dataset consists of listings of all the movies and tv shows available on Netflix, along with details such as - cast, directors, ratings, release year, duration, etc.

Data Storage

Documents are grouped
into an index



<input type="checkbox"/> Name	Health	Status	Primaries	Replicas	Docs count	Storage size
<input type="checkbox"/> portland_calendar	● green	open	1	1	2909086	142.25mb
<input type="checkbox"/> portland_listings	● green	open	1	1	7982	26.57mb
<input type="checkbox"/> portland_neighborhoods	● green	open	1	1	190	19.33kb
<input type="checkbox"/> reviews_portland	● green	open	1	1	628936	209.76mb
<input type="checkbox"/> netflix	● green	open	1	1	17610	9.42mb



```
{
  "_index" : "netflix",
  "_id" : "lxSAJ4EBrgs0t1RNYzAP",
  "_score" : 1.0,
  "_source" : {
    "country" : "United States",
    "show_id" : "s1",
    "director" : "Kirsten Johnson",
    "release_year" : 2020,
    "rating" : "PG-13",
    "description" : "As her father nears the end of his life
, filmmaker Kirsten Johnson stages his death in
inventive and comical ways to help them both face the
inevitable.",
    "type" : "Movie",
    "title" : "Dick Johnson Is Dead",
    "duration" : "90 min",
    "listed_in" : "Documentaries",
    "date_added" : "September 25, 2021"
  }
},
```



Data is stored as documents on Elasticsearch (a JSON
object under a unique ID)

Deployment: Elastic Cloud

Different ways to deploy



Self-Managed

Install a single package




Elastic Cloud

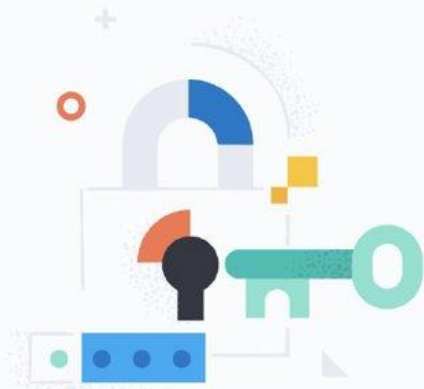
Deploy instantly on AWS,
Azure or Google Cloud



Using Elasticsearch



Elastic Cloud



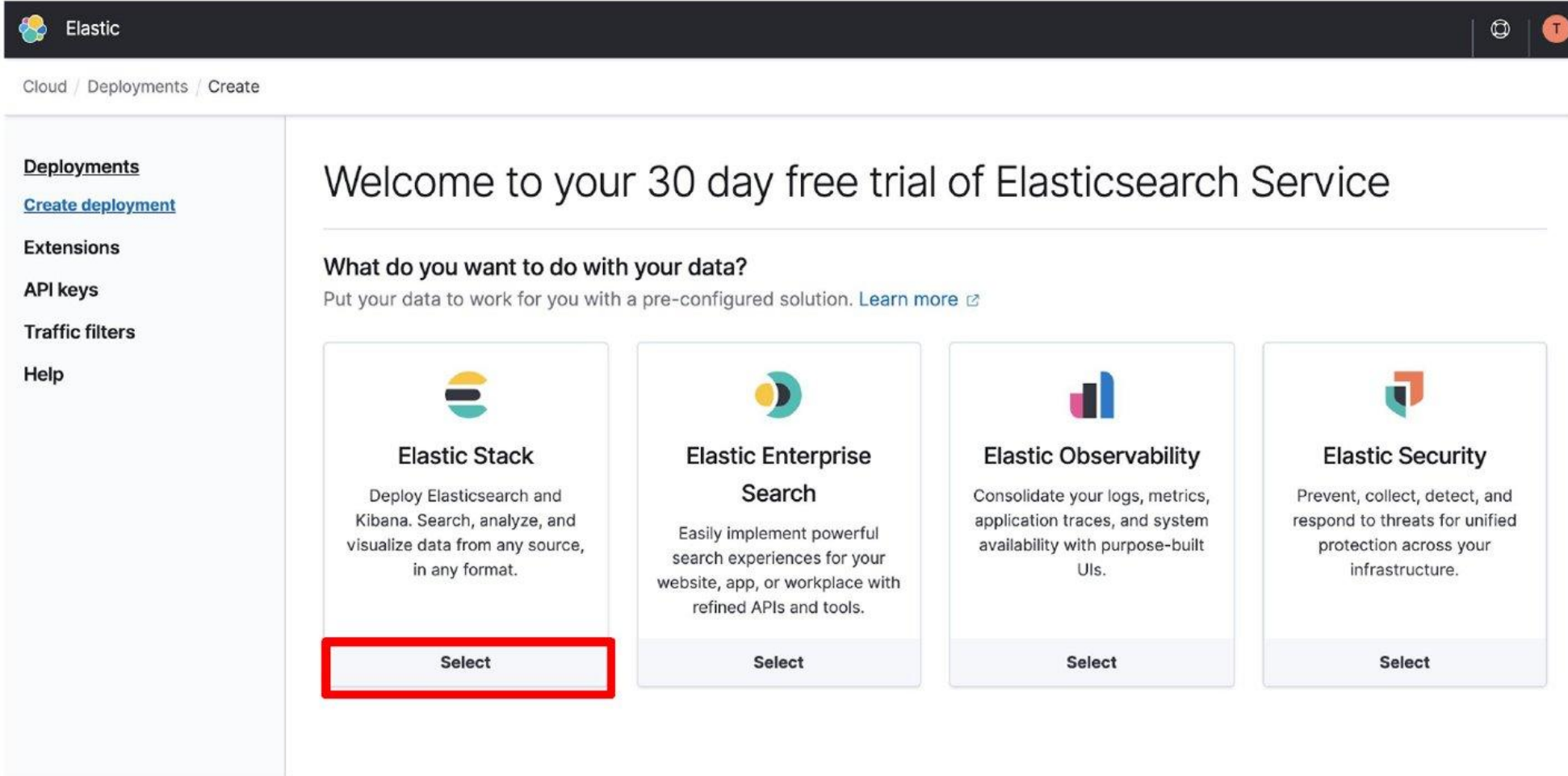
Set your password

Passwords must match the minimum security requirements. ?

New password

Set password and login

Select the Elastic Stack



The screenshot shows the Elastic Cloud interface for creating a new deployment. The left sidebar contains navigation links: Deployments, Create deployment, Extensions, API keys, Traffic filters, and Help. The main content area displays a welcome message for a 30-day free trial of Elasticsearch Service and asks the user what they want to do with their data. Four options are presented in cards: Elastic Stack, Elastic Enterprise Search, Elastic Observability, and Elastic Security. The 'Elastic Stack' card is highlighted with a red rectangular box, and its 'Select' button is also highlighted.

Elastic

Cloud / Deployments / Create

Deployments
[Create deployment](#)

Extensions


API keys

Traffic filters

Help

Welcome to your 30 day free trial of Elasticsearch Service


What do you want to do with your data?
Put your data to work for you with a pre-configured solution. [Learn more](#)



Elastic Stack

Deploy Elasticsearch and Kibana. Search, analyze, and visualize data from any source, in any format.


Select



Elastic Enterprise Search

Easily implement powerful search experiences for your website, app, or workplace with refined APIs and tools.


Select



Elastic Observability

Consolidate your logs, metrics, application traces, and system availability with purpose-built UIs.

Select



Elastic Security

Prevent, collect, detect, and respond to threats for unified protection across your infrastructure.

Select

Configure your settings

Select hardware profile

☒ **I/O Optimized** Recommended
Use for all-purpose workloads, including time-series data like logs and metrics. [See details](#)

☐ **Compute Optimized**
Run CPU-intensive workloads or run smaller workloads cost-effectively when you need less memory and storage. [See details](#)

☐ **Memory Optimized**
Perform memory-intensive operations efficiently, including workloads with frequent aggregations. [See details](#)

☐ **Hot-Warm Architecture**
Useful for time-series analytics that benefit from automatic index curation. [See details](#)

☐ **Cross Cluster Search** Not available in trial
Search data across one or more associated remote deployments. [See details](#)

Deployment settings
Choose the cloud provider, region, and Elastic Stack version.

Cloud provider
Pick a cloud and let us handle the rest. No additional accounts required.

Google Cloud

Azure

Amazon Web Services

Region
Select the location of your deployment.

West US 2 (Washington) ▾

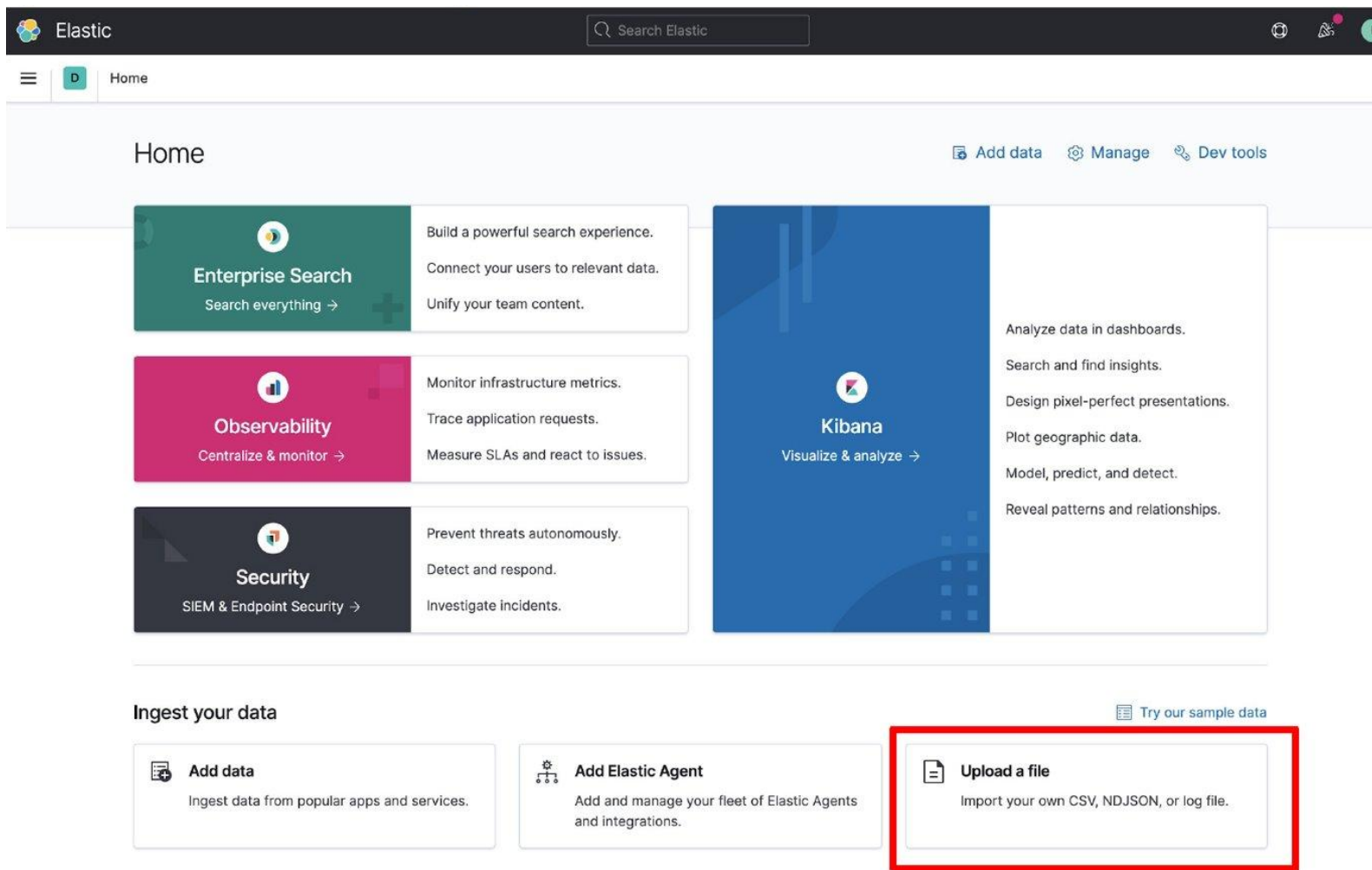
Version
Choose the Elastic Stack version.

7.10.1 ▾

Collapse ▾

Demonstration of Data Ingestion

File Upload



The screenshot displays the Elastic UI Home page. At the top, there is a dark header with the Elastic logo, a search bar labeled 'Search Elastic', and user profile icons. Below the header, a navigation bar shows 'Home' and links for 'Add data', 'Manage', and 'Dev tools'. The main content area is titled 'Home' and features three large cards: 'Enterprise Search' (green), 'Observability' (pink), and 'Security' (dark grey). To the right of these cards is a large blue card for 'Kibana'. Below the cards, the 'Ingest your data' section contains three options: 'Add data', 'Add Elastic Agent', and 'Upload a file'. The 'Upload a file' option is highlighted with a red rectangular border. It includes a document icon and the text 'Import your own CSV, NDJSON, or log file.'.

Home

[Add data](#) [Manage](#) [Dev tools](#)

Enterprise Search
Search everything →

Build a powerful search experience.
Connect your users to relevant data.
Unify your team content.

Observability
Centralize & monitor →

Monitor infrastructure metrics.
Trace application requests.
Measure SLAs and react to issues.

Security
SIEM & Endpoint Security →

Prevent threats autonomously.
Detect and respond.
Investigate incidents.

Kibana
Visualize & analyze →

Analyze data in dashboards.
Search and find insights.
Design pixel-perfect presentations.
Plot geographic data.
Model, predict, and detect.
Reveal patterns and relationships.

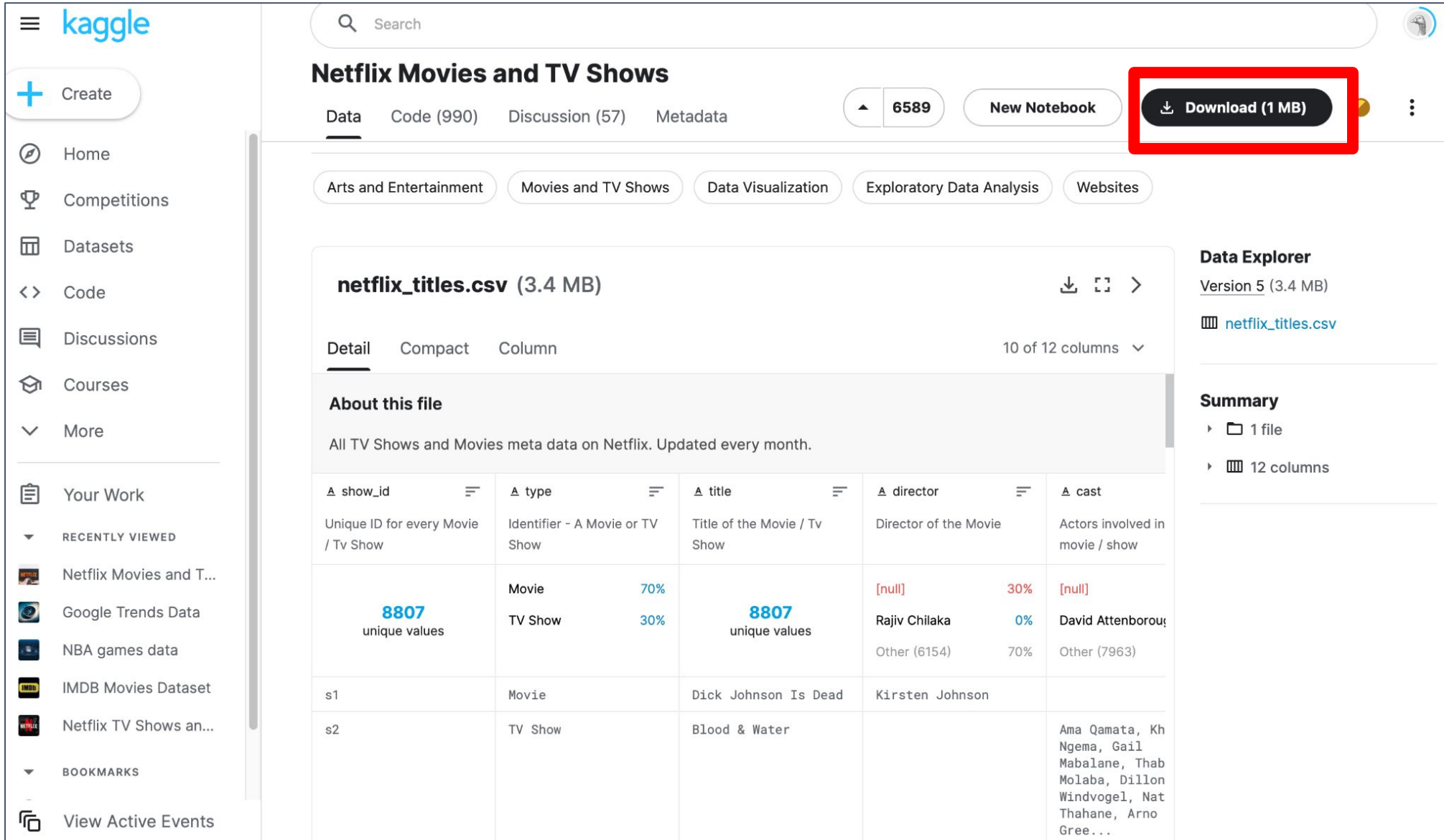
Ingest your data [Try our sample data](#)

Add data
Ingest data from popular apps and services.

Add Elastic Agent
Add and manage your fleet of Elastic Agents and integrations.

Upload a file
Import your own CSV, NDJSON, or log file.

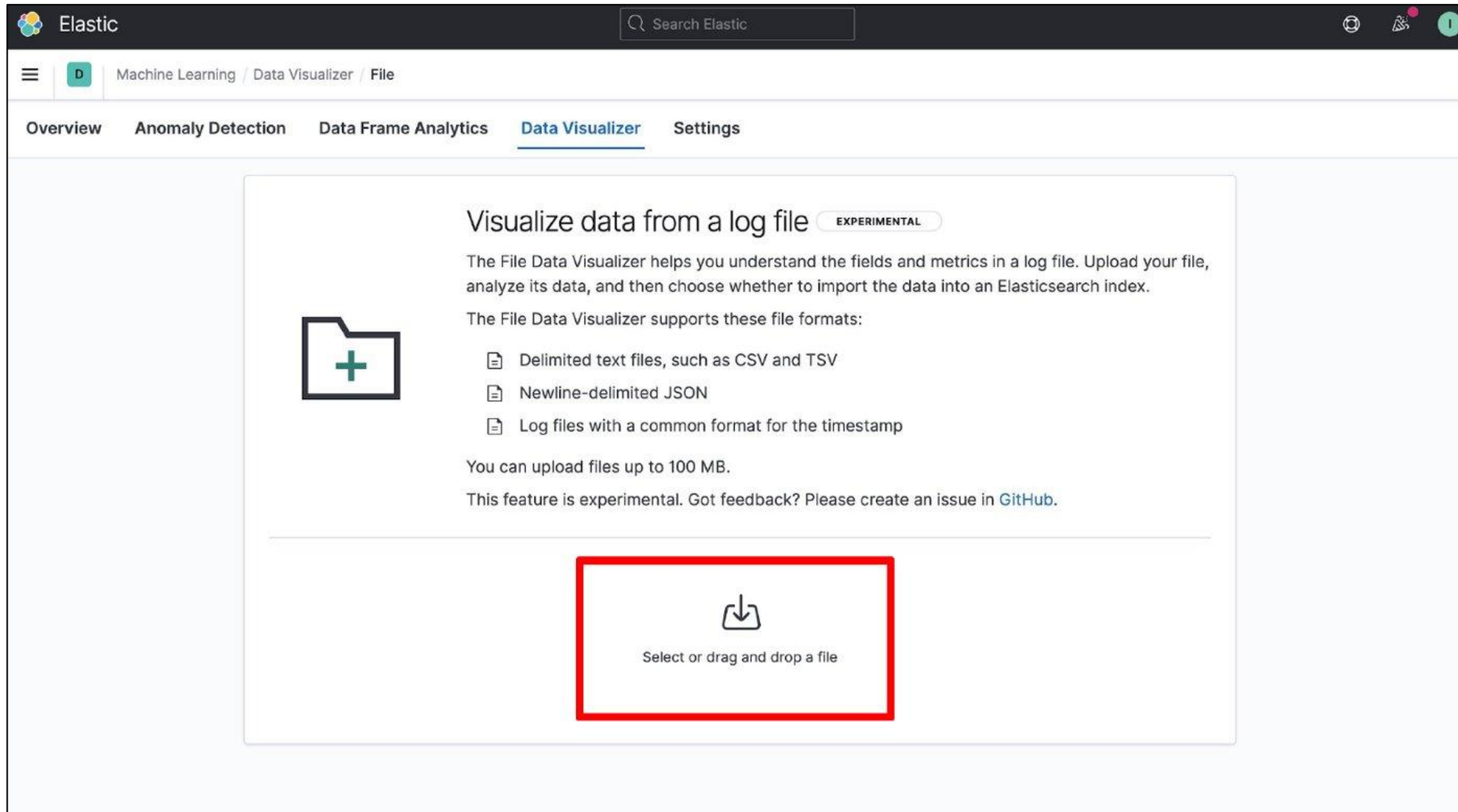
Download and unzip data from any source



The screenshot shows the Kaggle interface for the 'Netflix Movies and TV Shows' dataset. The 'Download (1 MB)' button is highlighted with a red box. The dataset is titled 'netflix_titles.csv (3.4 MB)' and is available in 'Data Explorer' version 5. The 'Summary' section indicates 1 file and 12 columns. The 'About this file' section describes the data as 'All TV Shows and Movies meta data on Netflix. Updated every month.' The table below shows the structure of the data, including columns for show_id, type, title, director, and cast.

show_id	type	title	director	cast
Unique ID for every Movie / Tv Show	Identifier - A Movie or TV Show	Title of the Movie / Tv Show	Director of the Movie	Actors involved in movie / show
8807 unique values	Movie 70% TV Show 30%	8807 unique values	[null] 30% Rajiv Chilaka 0% Other (6154) 70%	[null] David Attenborough Other (7963)
s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	
s2	TV Show	Blood & Water		Ama Qamata, Kh Ngema, Gail Mabalane, Thab Molaba, Dillon Windvogel, Nat Thahane, Arno Gree...

Drag and drop the file



The screenshot shows the Elastic Data Visualizer interface. The top navigation bar includes the Elastic logo, a search bar, and user profile icons. The main navigation tabs are Overview, Anomaly Detection, Data Frame Analytics, Data Visualizer (selected), and Settings. The Data Visualizer section is titled "Visualize data from a log file" with an "EXPERIMENTAL" badge. Below the title, a text block explains the File Data Visualizer's purpose: to help understand fields and metrics in a log file by uploading, analyzing, and optionally importing data into an Elasticsearch index. To the left of this text is a folder icon with a green plus sign. Below the text, a list of supported file formats is provided: Delimited text files (CSV, TSV), Newline-delimited JSON, and Log files with a common timestamp format. A note states that files up to 100 MB can be uploaded and provides a link to GitHub for feedback. At the bottom, a red rectangular box highlights a file upload area containing a download icon and the text "Select or drag and drop a file".

Elastic

Search Elastic

Machine Learning / Data Visualizer / File

Overview Anomaly Detection Data Frame Analytics **Data Visualizer** Settings

Visualize data from a log file EXPERIMENTAL


The File Data Visualizer helps you understand the fields and metrics in a log file. Upload your file, analyze its data, and then choose whether to import the data into an Elasticsearch index.

The File Data Visualizer supports these file formats:

- Delimited text files, such as CSV and TSV
- Newline-delimited JSON
- Log files with a common format for the timestamp

You can upload files up to 100 MB.

This feature is experimental. Got feedback? Please create an issue in [GitHub](#).



Select or drag and drop a file

Kibana will give you an analysis of the first 1000 lines of data

IntegrationsUpload file

netflix_titles.csv

File contents

First 1,000 lines

```
1 show_id,type,title,director,cast,country,date_added,release_year,rating,duration,listed_in,description
2 s1,Movie,Dick Johnson Is Dead,Kirsten Johnson,,United States,"September 25, 2021",2020,PG-13,90 min,Documentaries,"As her father nears the end of his
   life, filmmaker Kirsten Johnson stages his death in inventive and comical ways to help them both face the inevitable."
3 s2,TV Show,Blood & Water,,,"Ama Qamata, Khosi Ngema, Gail Mablane, Thabang Molaba, Dillon Windvogel, Natasha Thahane, Arno Greeff, Xolile Tshabalala,
   Getmore Sithole, Cindy Mahlangu, Ryle De Morny, Greteli Fincham, Sello Maake Ka-Ncube, Odwa Gwanya, Mekaila Mathys, Sandi Schultz, Duane Williams
   , Shamilla Miller, Patrick Mofokeng",South Africa,"September 24, 2021",2021,TV-MA,2 Seasons,"International TV Shows, TV Dramas, TV Mysteries"
   ,"After crossing paths at a party, a Cape Town teen sets out to prove whether a private-school swimming star is her sister who was abducted at
   birth."
4 s3,TV Show,Ganglands,Julien Leclercq,"Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabiha Akkari, Sofia Lesaffre, Salim Kechiouche, Noureddine Farihi,
   Geert Van Rampelberg, Bakary Diombera",,"September 24, 2021",2021,TV-MA,1 Season,"Crime TV Shows, International TV Shows, TV Action & Adventure"
   ,"To protect his family from a powerful drug lord, skilled thief Mehdi and his expert team of robbers are pulled into a violent and deadly turf
   war."
5 s4,TV Show,101birds New Orleans,,,"September 24, 2021",2021,TV-MA,1 Season,"Documentaries, Reality TV","Faude, flirtations and toilet talk go down
```

Summary

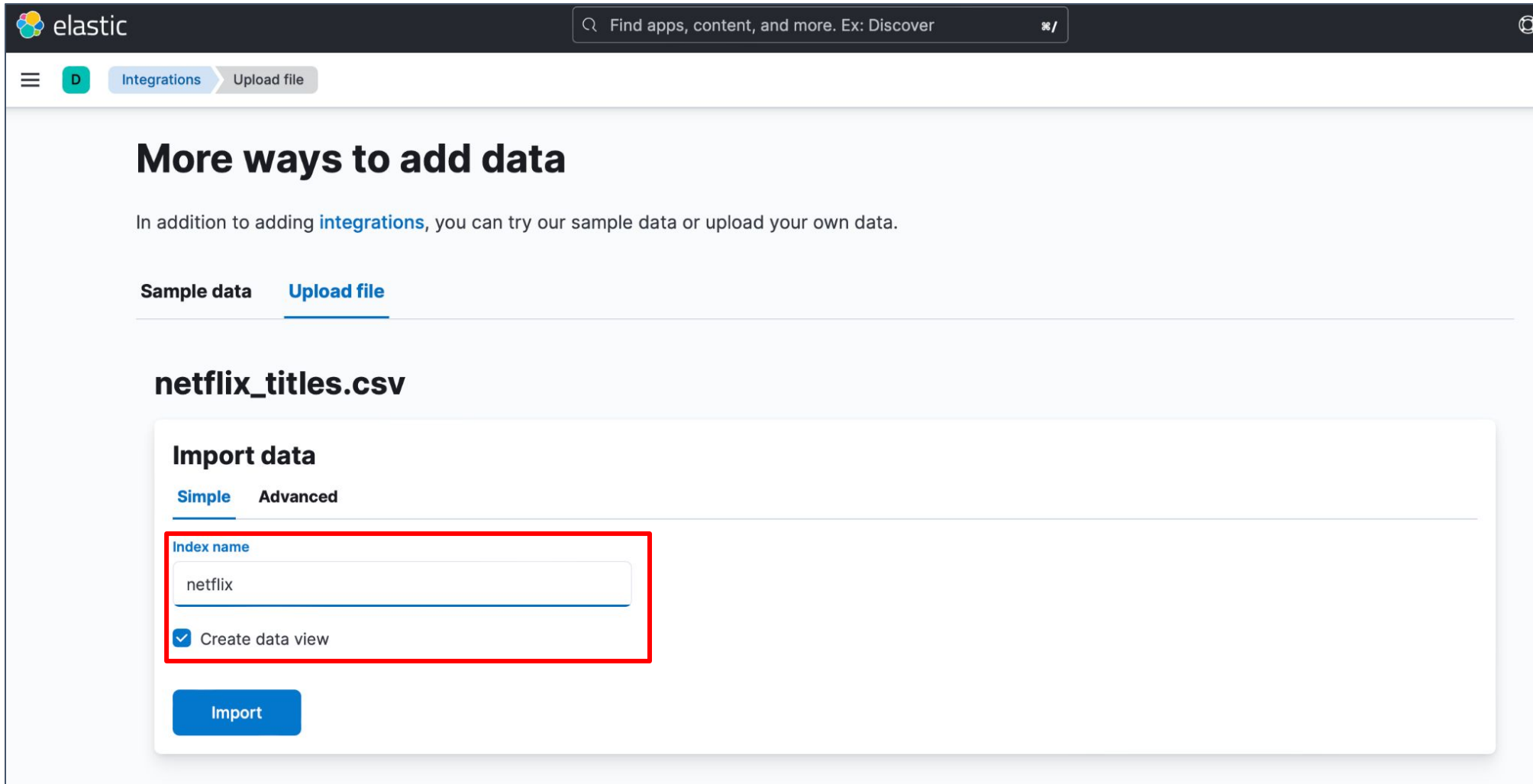
Number of lines analyzed	1000
Format	delimited
Delimiter	,
Has header row	true

[Override settings](#) [Analysis explanation](#)

Import

Cancel

Name your index and click on import



The screenshot shows the Elastic UI interface for uploading data. The top navigation bar includes the Elastic logo, a search bar with the text "Find apps, content, and more. Ex: Discover", and a user icon. The main navigation bar shows a menu icon, a "D" icon, and two tabs: "Integrations" and "Upload file". The "Upload file" tab is active.

More ways to add data

In addition to adding [integrations](#), you can try our sample data or upload your own data.

Sample data **Upload file**

netflix_titles.csv

Import data

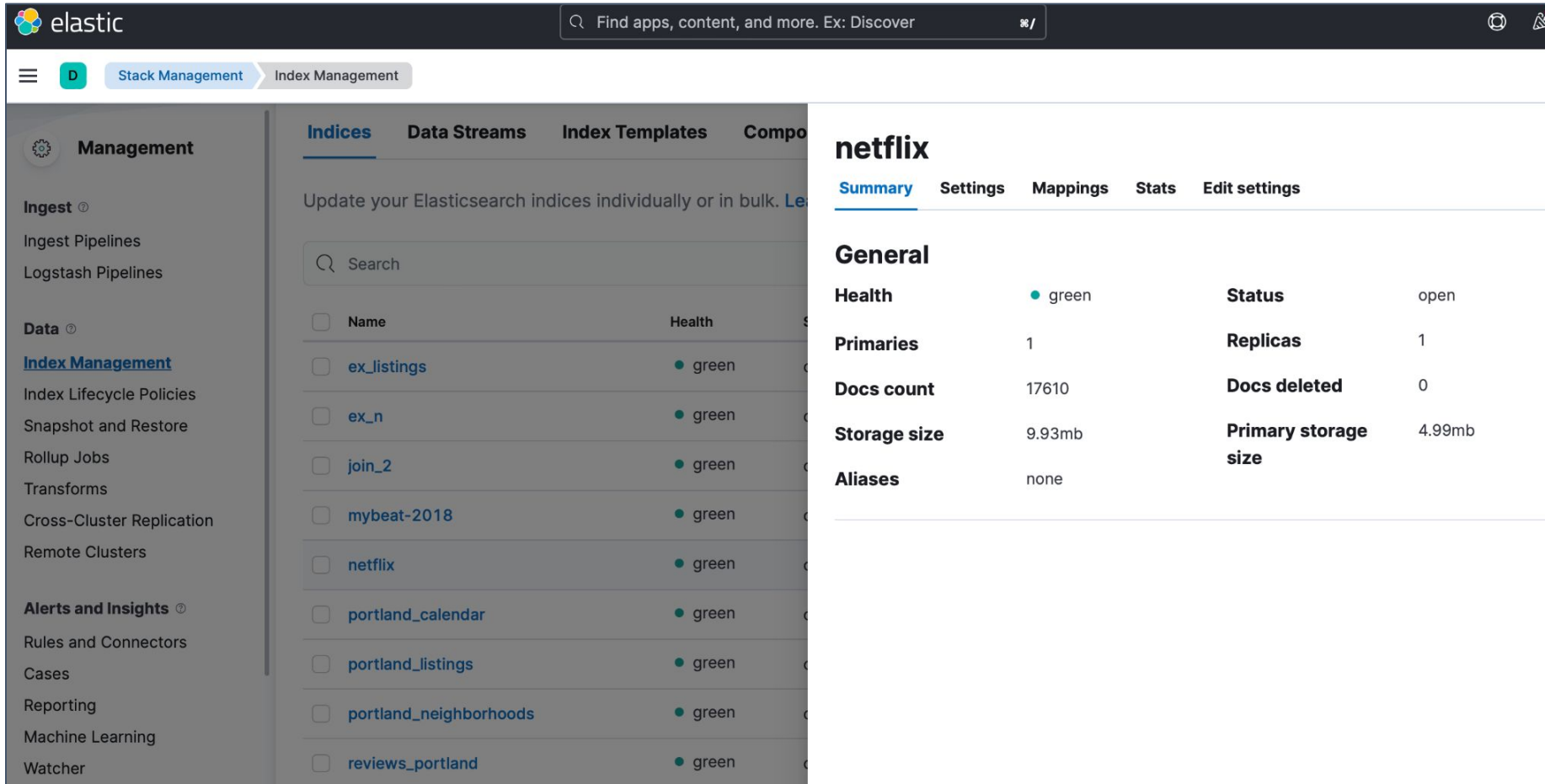
Simple **Advanced**

Index name

☒ Create data view

Import

View/Manage your index



The screenshot shows the Elastic Stack Management interface. The top navigation bar includes the Elastic logo, a search bar, and icons for help and notifications. The main navigation menu on the left is divided into 'Management' and 'Data' sections. The 'Management' section includes 'Ingest' (Ingest Pipelines, Logstash Pipelines) and 'Alerts and Insights' (Rules and Connectors, Cases, Reporting, Machine Learning, Watcher). The 'Data' section includes 'Index Management' (Index Lifecycle Policies, Snapshot and Restore, Rollup Jobs, Transforms, Cross-Cluster Replication, Remote Clusters). The 'Index Management' page is active, showing a list of indices under the 'Indices' tab. The list includes indices like 'ex_listings', 'ex_n', 'join_2', 'mybeat-2018', 'netflix', 'portland_calendar', 'portland_listings', 'portland_neighborhoods', and 'reviews_portland', all with a 'green' health status. The 'netflix' index is selected, and its details are shown on the right. The details page has tabs for 'Summary', 'Settings', 'Mappings', 'Stats', and 'Edit settings'. The 'Summary' tab is active, showing 'General' information: Health (green), Status (open), Primaries (1), Replicas (1), Docs count (17610), Docs deleted (0), Storage size (9.93mb), Primary storage size (4.99mb), and Aliases (none).

elastic

Find apps, content, and more. Ex: Discover

Stack Management Index Management

Management

Ingest

- Ingest Pipelines
- Logstash Pipelines

Data

Index Management

- Index Lifecycle Policies
- Snapshot and Restore
- Rollup Jobs
- Transforms
- Cross-Cluster Replication
- Remote Clusters

Alerts and Insights

- Rules and Connectors
- Cases
- Reporting
- Machine Learning
- Watcher

Indices Data Streams Index Templates Components

Update your Elasticsearch indices individually or in bulk. [Learn more](#)

Search

<input type="checkbox"/>	Name	Health
<input type="checkbox"/>	ex_listings	green
<input type="checkbox"/>	ex_n	green
<input type="checkbox"/>	join_2	green
<input type="checkbox"/>	mybeat-2018	green
<input type="checkbox"/>	netflix	green
<input type="checkbox"/>	portland_calendar	green
<input type="checkbox"/>	portland_listings	green
<input type="checkbox"/>	portland_neighborhoods	green
<input type="checkbox"/>	reviews_portland	green

netflix

[Summary](#) [Settings](#) [Mappings](#) [Stats](#) [Edit settings](#)

General

Health	green	Status	open
Primaries	1	Replicas	1
Docs count	17610	Docs deleted	0
Storage size	9.93mb	Primary storage size	4.99mb
Aliases	none		

Query Execution

Query 1

```
# Query 1: Matching
# From the Netflix dataset, display the title, release year and the rating for
# movies which have the words "Nazi" OR "Camps" but NOT "deaths" in the title.
# Sort the responses in the descending order of release year.
```

GET netflix/_search

```
{
  "_source": ["title", "release_year", "rating"],
  "sort": [
    {
      "release_year": {
        "order": "desc"
      }
    }
  ],
  "query": {
    "bool": {
      "should": [
        {
          "match": {
            "title": "Nazi"
          }
        },
        {
          "match": {
            "title": "Camps"
          }
        }
      ],
      "must_not": {
        "match": {
          "title": "Death"
        }
      }
    }
  }
}
```

```
1 {
2   "took" : 0,
3   "timed_out" : false,
4   "_shards" : {
5     "total" : 1,
6     "successful" : 1,
7     "skipped" : 0,
8     "failed" : 0
9   },
10  "hits" : {
11    "total" : {
12      "value" : 3,
13      "relation" : "eq"
14    },
15    "max_score" : null,
16    "hits" : [
17      {
18        "_index" : "netflix",
19        "_id" : "txZH04EBrgs0t1RNILj2",
20        "_score" : null,
21        "_source" : {
22          "release_year" : 2016,
23          "rating" : "TV-PG",
24          "title" : "Nazi Mega Weapons"
25        },
26        "sort" : [
27          2016
28        ]
29      },
30      {
31        "_index" : "netflix",
32        "_id" : "2ANH04EBSwXPRGhyHexm",
33        "_score" : null,
```

Query 2

```
# Query 2: Nested Boolean
# Display the title of all the movies in the Netflix dataset where the title
# contains the word "Nazi" OR "King" and DOES NOT contain the words "Girl" OR
# "Boys". Sort them in the ascending order of year.
```

```
GET netflix/_search
```

```
{
  "_source": ["title"],
  "sort": [
    {
      "release_year": {
        "order": "asc"
      }
    }
  ],
  "query": {
    "bool": {
      "should": [
        {
          "match": {
            "title": "King"
          }
        },
        {
          "match": {
            "title": "Nazi"
          }
        }
      ],
      "must_not": [
        {
          "bool": {
            "should": [
              {
                "match": {
```

```
          "title": "Girl"
            },
            {
              "match": {
                "title": "Boys"
              }
            }
          ]
        }
      ]
    }
  }
}
```

```
1 {
2   "took" : 1,
3   "timed_out" : false,
4   "_shards" : {
5     "total" : 1,
6     "successful" : 1,
7     "skipped" : 0,
8     "failed" : 0
9   },
10  "hits" : {
11    "total" : {
12      "value" : 37,
13      "relation" : "eq"
14    },
15    "max_score" : null,
16    "hits" : [
17      {
18        "_index" : "netflix",
19        "_id" : "thZH04EBrgs0t1RNILj2",
20        "_score" : null,
21        "_source" : {
22          "title" : "Nazi Concentration Camps"
23        },
24        "sort" : [
25          1945
26        ]
27      },
28      {
29        "_index" : "netflix",
30        "_id" : "5gNH04EBSwXPRGhyHetm",
31        "_score" : null,
32        "_source" : {
33          "title" : "Joseph: King of Dreams"
```

Query 3

```
# Query 3: multi-field and boosting
# Write a query to search a phrase from multiple text fields in the
Netflix index. Display the search results of the phrase "family"
in cast, listed_in and description. Also prioritize the search
results of the phrase appearing in the genre category (field name:
listed_in) first.
```

```
GET netflix/_search
```

```
{
  "_source": ["title", "description", "listed_in"],
  "query": {
    "multi_match": {
      "query": "family",
      "fields": [
        "description",
        "cast",
        "listed_in^2"
      ]
    }
  }
}
```



||

```
10- "hits" : {
11-   "total" : {
12-     "value" : 1163,
13-     "relation" : "eq"
14-   },
15-   "max_score" : 5.959728,
16-   "hits" : [
17-     {
18-       "_index" : "netflix",
19-       "_id" : "MwNH04EBSwXPRGhyEd7g",
20-       "_score" : 5.959728,
21-       "_source" : {
22-         "listed_in" : "Children & Family Movies",
23-         "description" : "Equestria's divided. But a bright-eyed hero believes
Earth Ponies, Pegasi and Unicorns should be pals - and, hoof to heart,
she's determined to prove it.",
24-         "title" : "My Little Pony: A New Generation"
25-       }
26-     },
27-     {
28-       "_index" : "netflix",
29-       "_id" : "RANH04EBSwXPRGhyEd7g",
30-       "_score" : 5.959728,
31-       "_source" : {
32-         "listed_in" : "Children & Family Movies",
33-         "description" : "From arcade games to sled days and hiccup cures, Cory
Carson's curious little sister Chrissy speeds off on her own for fun
and adventure all over town!",
34-         "title" : "Go! Go! Cory Carson: Chrissy Takes the Wheel"
35-       }
36-     },
37-     {
38-       "_index" : "netflix",
```


Query 4

Query 4: Fuzziness function

Write a query to incorporate user error like spelling mistakes in their searches. For example misspelling the word fiction in description. Return the title, description and cast with the result ordered in ascending order of release year and limit the result to first 10 rows.

POST netflix/_search

```
{
  "sort": [
    {
      "release_year": {
        "order": "asc"
      }
    }
  ],
  "_source": ["title", "description", "cast"],
  "query": {
    "multi_match": {
      "query": "science fction",
      "fields": ["description"],
      "fuzziness": 2
    }
  },
  "size": 10
}
```

```
1 {
2   "took" : 8,
3   "timed_out" : false,
4   "_shards" : {
5     "total" : 1,
6     "successful" : 1,
7     "skipped" : 0,
8     "failed" : 0
9   },
10  "hits" : {
11    "total" : {
12      "value" : 202,
13      "relation" : "eq"
14    },
15    "max_score" : null,
16    "hits" : [
17      {
18        "_index" : "netflix",
19        "_id" : "JQNH04EBSwXPRGhyHe9m",
20        "_score" : null,
21        "_source" : {
22          "cast" : "Leslie Nielsen, Walter Pidgeon, Anne Francis, Jack Kelly,
23            Warren Stevens, James Drury, Earl Holliman",
24          "description" : "A pulp-fiction sci-fi classic, Forbidden Planet stars
25            Leslie Nielsen as a heroic starship captain who finds the paradise
26            planet Altair-IV.",
27          "title" : "Forbidden Planet"
28        },
29        "sort" : [
30          1956
31        ]
32      }
33    ]
34  }
35}
```

Query 5

Query 5: Flexible searches - Slop functionality

Write a query to create more flexible searches. Write a query to search for "time travel" in the description field using the slop parameter of allowing skip of 1 word. Eg "time trval" can be matched by "time for travel" document as well. Display the title and description of the result.

GET netflix/_search

```
{
  "_source": ["title", "description"],
  "query": {
    "match_phrase": {
      "description": {
        "query": "time travel",
        "slop": 1
      }
    }
  }
}
```

```
25 ~ },
26 ~ {
27   "_index" : "netflix",
28   "_id" : "ORZH04EBrgs0t1RNGLQZ",
29   "_score" : 8.3662405,
30 ~   "_source" : {
31     "description" : "As two teen prodigies try to master the art of time
      travel, a tragic police shooting sends them on a series of dangerous
      trips to the past.",
      "title" : "See You Yesterday"
32   }
33 ~ }
34 ~ },
35 ~ {
36   "_index" : "netflix",
37   "_id" : "DxZH04EBrgs0t1RNILr2",
38   "_score" : 8.3662405,
39 ~   "_source" : {
40     "description" : "A team of journalists are sent to find out who's behind a
      classified ad seeking a companion for time travel, then must decide how
      to approach him.",
      "title" : "Safety Not Guaranteed"
41   }
42 ~ }
43 ~ },
44 ~ {
45   "_index" : "netflix",
46   "_id" : "YxZH04EBrgs0t1RNIL32",
47   "_score" : 5.3423214,
48 ~   "_source" : {
49     "description" : "A man cured of leprosy and a young orphan leave their
      colony for the first time to travel across Egypt, hoping to find the
      families they lost.",
50     "title" : "يوم الدين"
51 ~   }
}
```

Query 6

```
# Query 6: Release trend over the years
# Display the number of movies/TV series released throughout the
  years(write a query for time series indication using aggregations)
```

```
GET /netflix/_search
{
  "aggs": {
    "by_category": {
      "multi_terms": {
        "terms": [{
          "field": "type"
        }, {
          "field": "release_year"
        }]
      }
    }
  }
}
```

```
1 {
2   "took" : 11,
3   "timed_out" : false,
4   "_shards" : {
5     "total" : 1,
6     "successful" : 1,
7     "skipped" : 0,
8     "failed" : 0
9   },
10  "hits" : { },
200 "aggregations" : {
201   "by_category" : {
202     "doc_count_error_upper_bound" : 0,
203     "sum_other_doc_count" : 3538,
204     "buckets" : [
205       {
206         "key" : [
207           "Movie",
208           2017
209         ],
210         "key_as_string" : "Movie|2017",
211         "doc_count" : 767
212       },
213       {
214         "key" : [
215           "Movie",
216           2018
217         ],
218         "key_as_string" : "Movie|2018",
219         "doc_count" : 767
220       },
221       {
222         "key" : [
```


Query 7

```
# Query 7: Querying and Aggregation
# Display the most common/ significant terms used while providing a
  brief description of a movie/TV show.
```

```
GET netflix/_search
{
  "query": {
    "match": {
      "type": "TV Show"
    }
  },
  "aggregations": {
    "popular_in_tv_shows": {
      "significant_text": {
        "field": "listed_in"
      }
    }
  }
}
```

```
1 {
2   "took" : 41,
3   "timed_out" : false,
4   "_shards" : {
5     "total" : 1,
6     "successful" : 1,
7     "skipped" : 0,
8     "failed" : 0
9   },
10  "hits" : { },
194  "aggregations" : {
195    "popular_in_tv_shows" : {
196      "doc_count" : 2676,
197      "bg_count" : 8805,
198      "buckets" : [
199        {
200          "key" : "tv",
201          "doc_count" : 2571,
202          "score" : 2.200490408011422,
203          "bg_count" : 2571
204        },
205        {
206          "key" : "shows",
207          "doc_count" : 1793,
208          "score" : 1.5346088298578295,
209          "bg_count" : 1793
210        },
211        {
212          "key" : "crime",
213          "doc_count" : 470,
214          "score" : 0.4022677914295481,
215          "bg_count" : 470
216        },

```

Lessons Learned

- Provides high performance
- Strived for fine-tuning and increasing the relevance of the search
- Is Easily scalable
- Distributed Architecture
- Document-oriented Database
- Schema Free

References

- [Elastic Official Website](#)
- [Elasticsearch Guide \[8.2\] | documents and indices](#)
- [Elasticsearch: What it is, How it works, and what it's used for](#)
- [Can Elasticsearch Store Data?](#)
- [Elasticsearch: Guide | Scalability](#)
- [Scaling Elasticsearch | Medium](#)
- [Is Elasticsearch SET/GET Eventually Consistent? - DZone Big Data](#)
- [An overview on elasticsearch and its usage](#)
- [Significant text aggregation](#)
- [SQL Limitations | Elasticsearch Guide \[8.2\] | Elastic](#)
- [Nested query | Elasticsearch Guide \[8.2\] | Elastic](#)
- [Querying multiple Elasticsearch indexes with different structures both nested and 'plain' - Stack Overflow](#)
- [Elastic Resource](#)

Thank You