CIS 3200 Term Project Tutorial Group 3

Authors: Vanessa Munoz; Shawn Tran; Hae-In Kim; David Montes

Instructor: Jongwook Woo

Date: 05/11/2021

Lab Tutorial

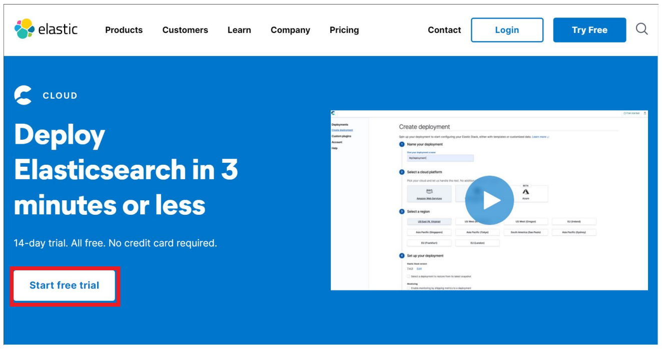
Objectives

In this Hands-on lab you will learn how to: Visualize a dataset through ElasticSearch and Kibana and create AzureML experiments to predict data.

Step 1: Creating an account on Elastic Cloud and logging in

In this first step we will go through signing up and setting up an Elastic Cloud

1. Visit https://www.elastic.co/ and click the **Start free trial** button.



Deploy Elasticsearch in 3 minutes or less

14-day trial. All free. No credit card required.

2. Create your account using an existing email.

3. Click on the verification link in the email you provided.

4. Follow the prompts and proceed to login to Elastic Cloud.

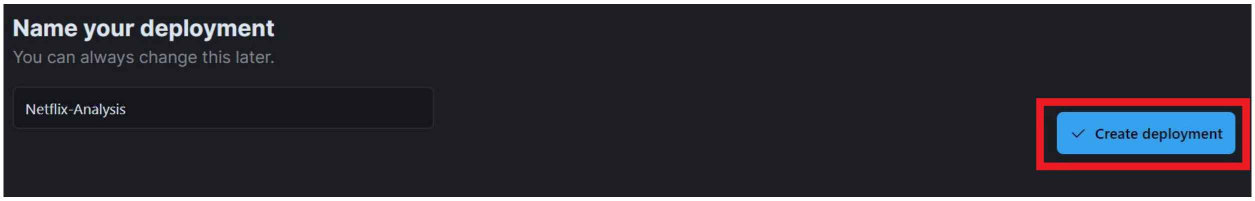
Step 2: Creating your first hosted Elastic Search Cluster

1. After signing in, click the **Start your free trial** button to begin creating a deployment.

2. Click **Select** under the General Purpose box.

3. Give your deployment a name.

    a. In this case we have named our deployment *Netflix-Analysis.*



Name your deployment

You can always change this later.

Netflix-Analysis

4. Select **Create deployment**.

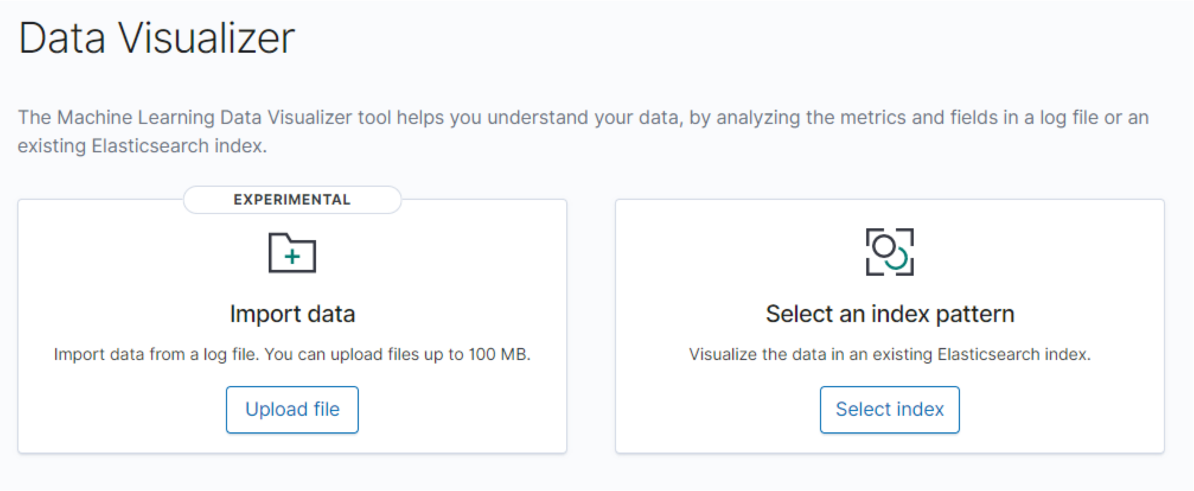
5. Take note of the username and password for this deployment.

6. After a few minutes once the deployment has been created click **Open Kibana**.

7. Click **Explore on my own**.

Step 3: Adding data to Elastic Cloud and index mapping

1. Click the data visualizer tab under the machine learning section of Kibana.



Data Visualizer

The Machine Learning Data Visualizer tool helps you understand your data, by analyzing the metrics and fields in a log file or an existing Elasticsearch index.

Import data

Import data from a log file. You can upload files up to 100 MB.

Select an index pattern

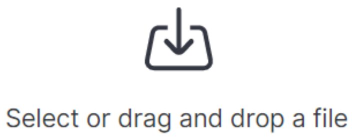
Visualize the data in an existing Elasticsearch index.

2. Open a new browser tab or new browser window and navigate to the following link:

https://github.com/stran13/CIS3200Netflix

3. Click the netflix_titles.csv file and download it.

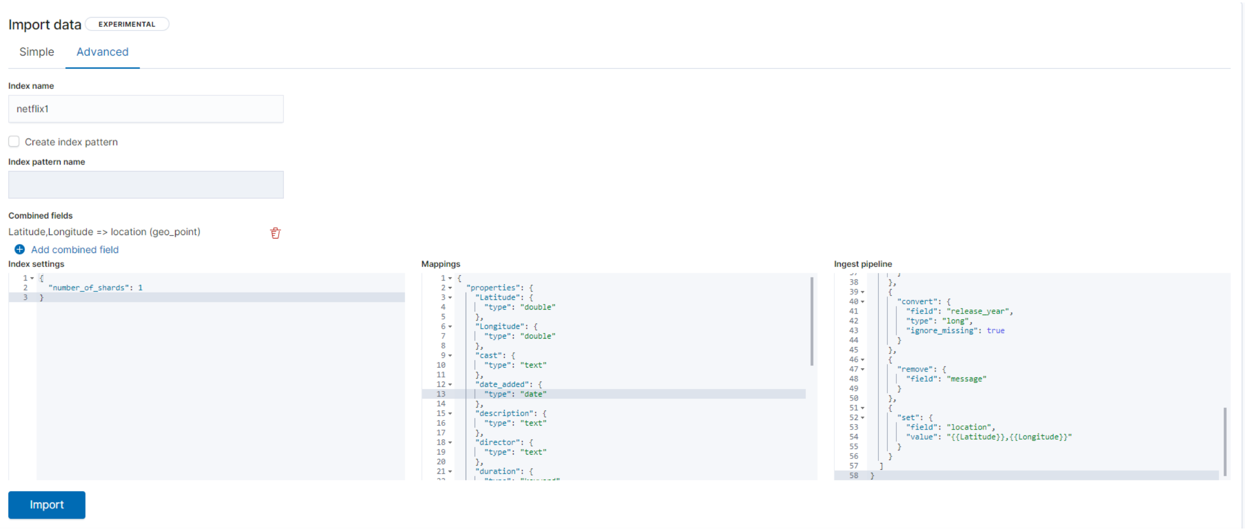
4. Go back to the previous browser window, drag the **netflix_titles.csv** to the icon shown below:



Select or drag and drop a file

5. On the next screen, click **Import**.

6. Select the advanced tab and uncheck the **Create index pattern** button. Then name the index anything you want, in this case we named it netflix1. Then under the mappings section, change the type of **date_added** to date. Make sure your screen looks like this and then click import.



Import data EXPERIMENTAL

Simple Advanced

Index name: netflix1

□ Create index pattern

Index pattern name:

Combined fields

Latitude,Longitude => location (geo_point)

Add combined field

Index settings

```
{
  "number_of_shards": 1
}
```

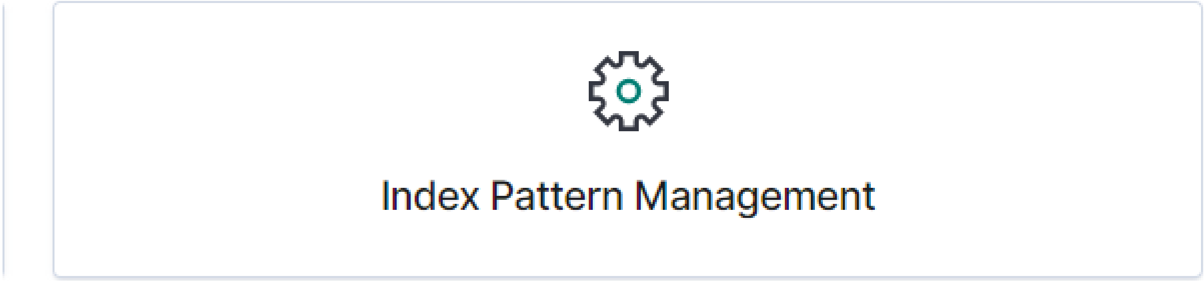
Mappings

```
{
  "properties": {
    "Latitude": {
      "type": "double"
    },
    "Longitude": {
      "type": "double"
    },
    "cast": {
      "type": "text"
    },
    "date_added": {
      "type": "date"
    },
    "description": {
      "type": "text"
    },
    "director": {
      "type": "text"
    },
    "duration": {
```

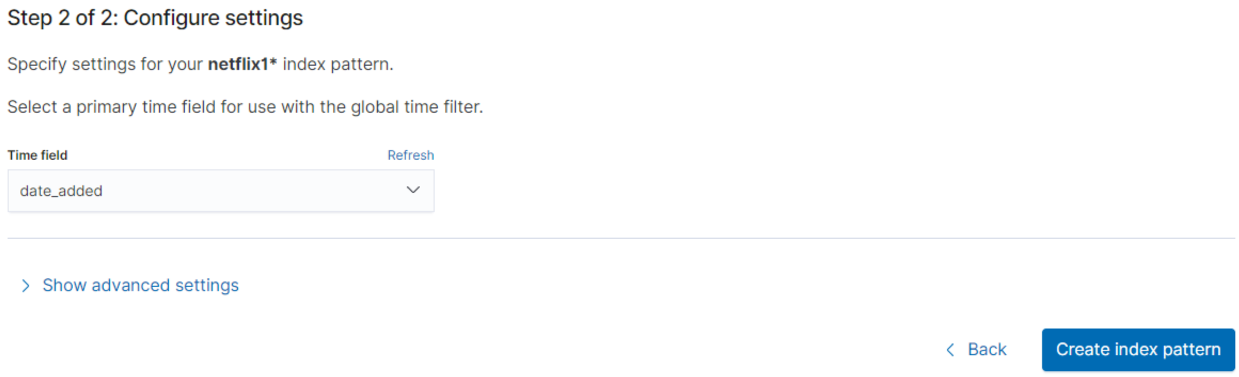
Ingest pipeline

```
    },
    {
      "convert": {
        "field": "release_year",
        "type": "long",
        "ignore_missing": true
      }
    },
    {
      "remove": {
        "field": "message"
      }
    },
    {
      "set": {
        "field": "location",
        "value": "{{Latitude}},{{Longitude}}"
      }
    }
  ]
}
```

7. After the file has been imported click the index pattern management button at the bottom of the screen.



8. Once here, click the create index pattern button and type the name of the index you created previously. Then click next step. Select date_added under the Time field drop down menu and select create index pattern.



Step 2 of 2: Configure settings

Specify settings for your **netflix1*** index pattern.

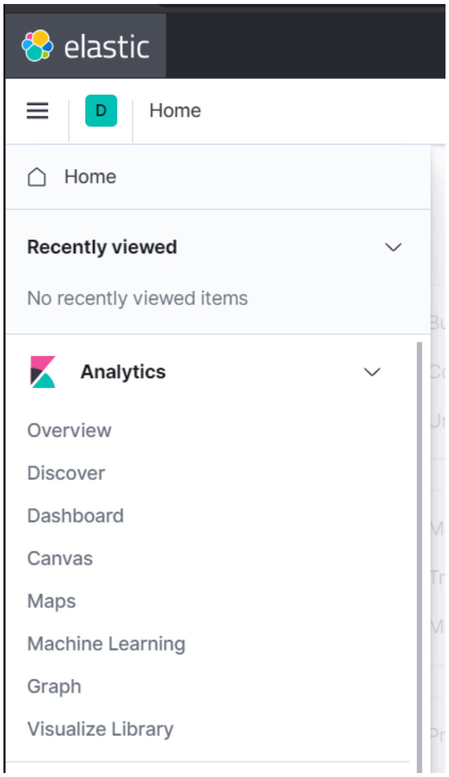
Select a primary time field for use with the global time filter.

Time field: date_added

Step 4: Creating a Bar Graph

1. Click on the 3 lines in the top left corner to bring up the side bar.

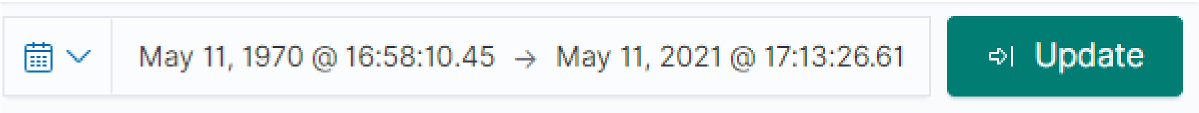
2. Click on **Visualize Library**.



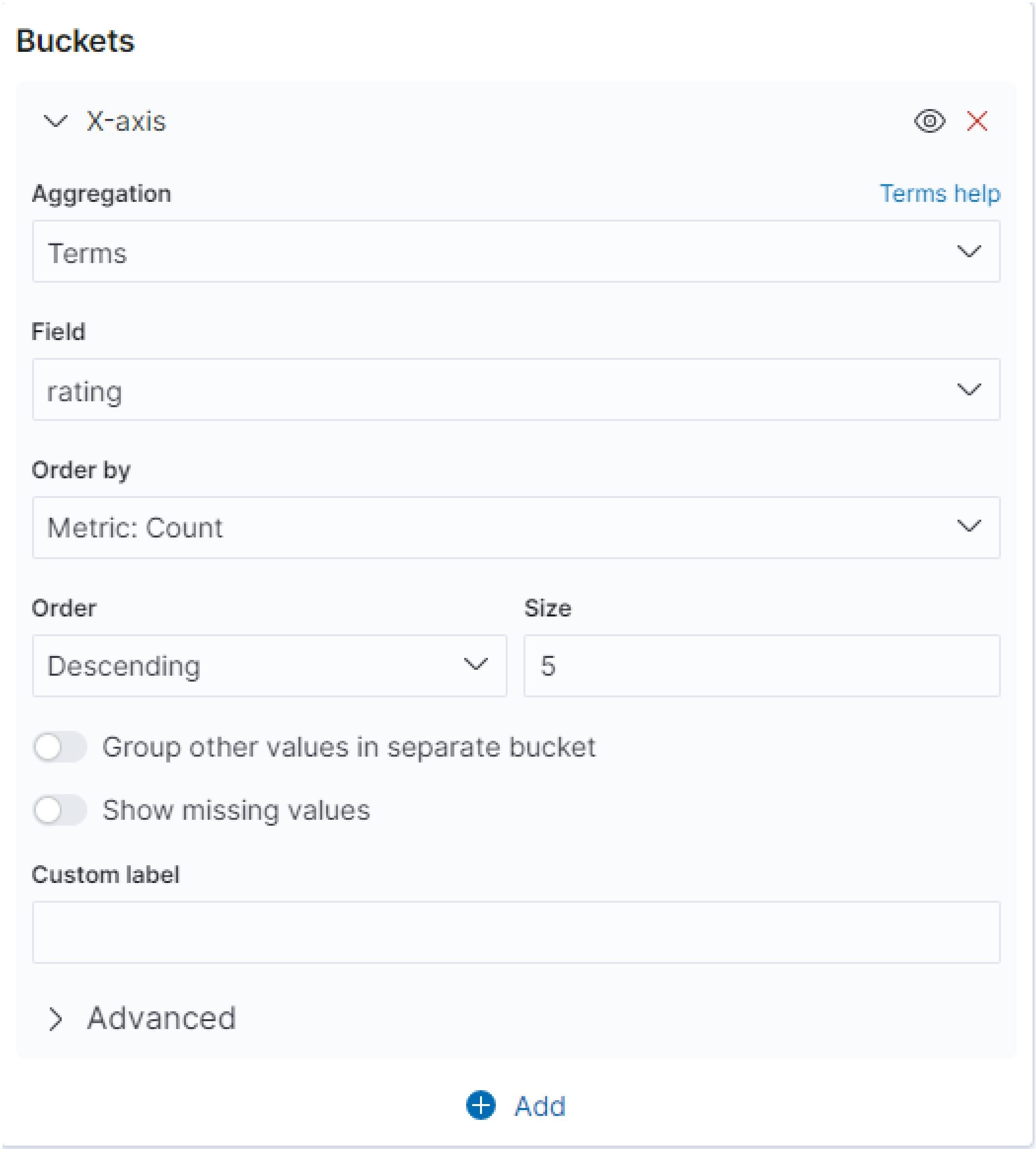
3. Click on **Create new visualization.**

4. Click on **Vertical bar** and type the name of the created index pattern and select it.

5. Change the date at the top to absolute from 1970 to today's date and click update.


May 11, 1970 @ 16:58:10.45 → May 11, 2021 @ 17:13:26.61

6. Click the add button under the buckets section on the right panel. Select X-axis and configure the bucket as so and click update.



Buckets

X-axis

Aggregation: Terms

Field: rating

Order by: Metric: Count

Order: Descending

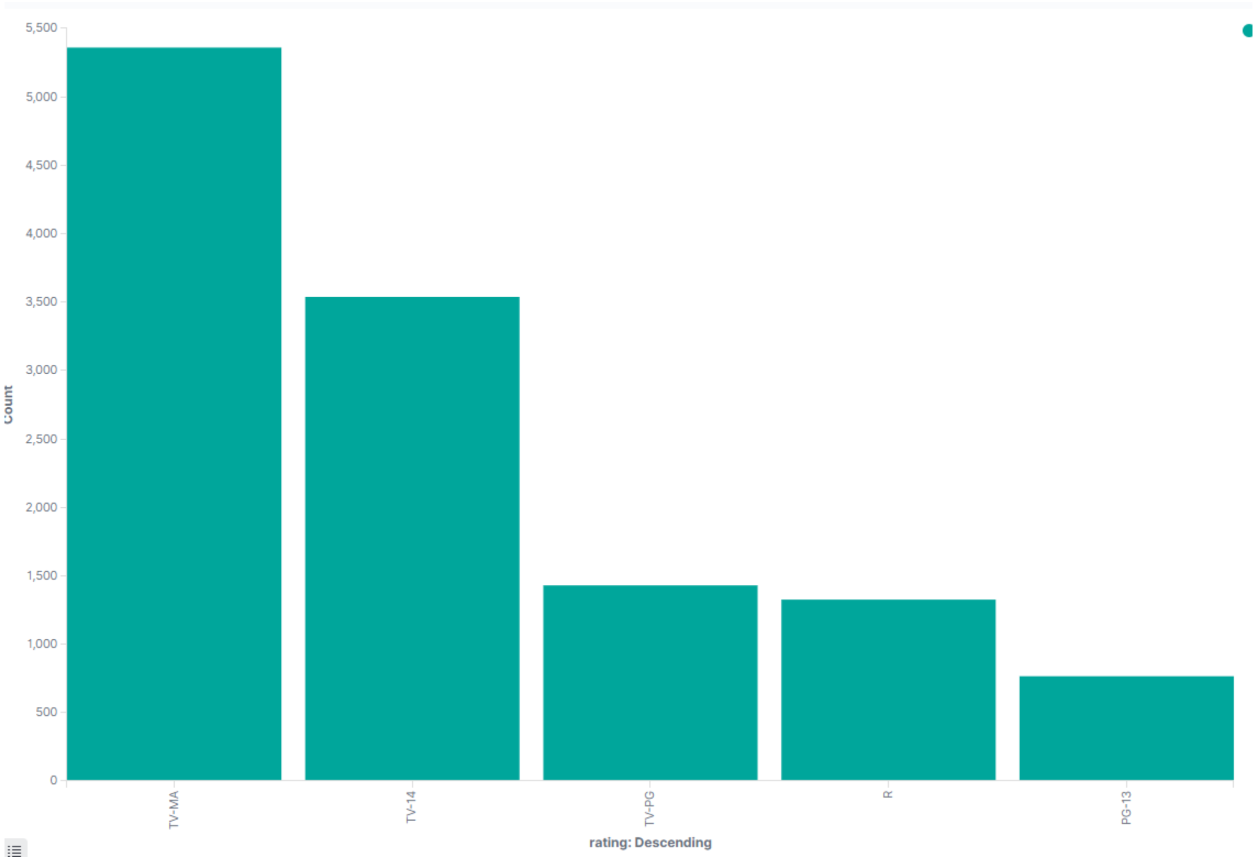
Size: 5

○ Group other values in separate bucket

○ Show missing values

Custom label

7. The bar chart should look something like this.



8. Save the bar chart as **netflixbar**.

Step 5: Creating a Pie Chart

1. Click on the 3 lines in the top left corner to bring up the side bar.

2. Click on **Visualize** again.

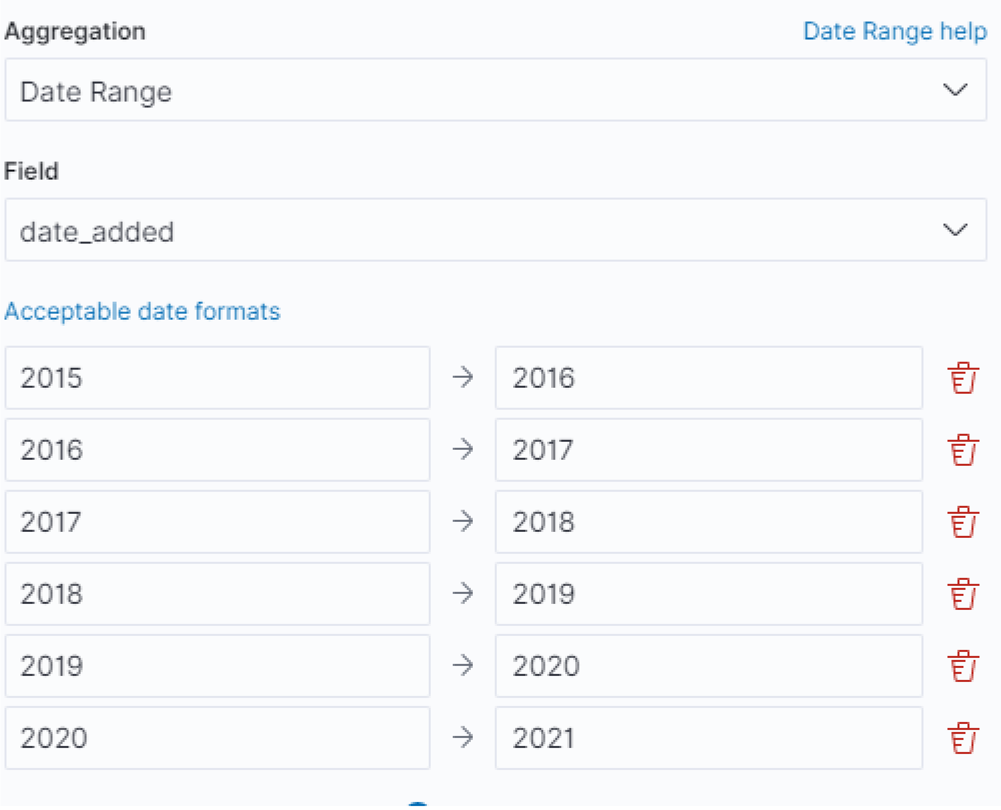
3. Click on **Create new visualization.**

4. Click on **Pie**.

5. Search the index pattern you created and select it.

6. Make sure the date at the top right matches the previous dates on the bar graph.

7. Click add buckets and select split slices. And configure the bucket as so and click update.



Aggregation

Date Range help

Date Range

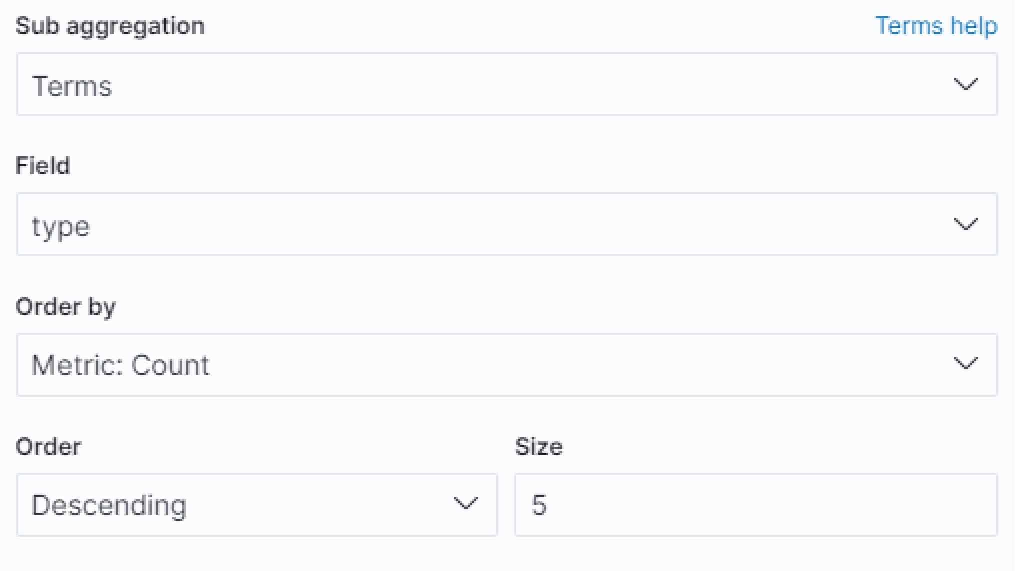
Field

date_added

Acceptable date formats

2015	→	2016
2016	→	2017
2017	→	2018
2018	→	2019
2019	→	2020
2020	→	2021

8. Then create another split slices and configure it as so and click update.



Sub aggregation

Terms help

Terms

Field

type

Order by

Metric: Count

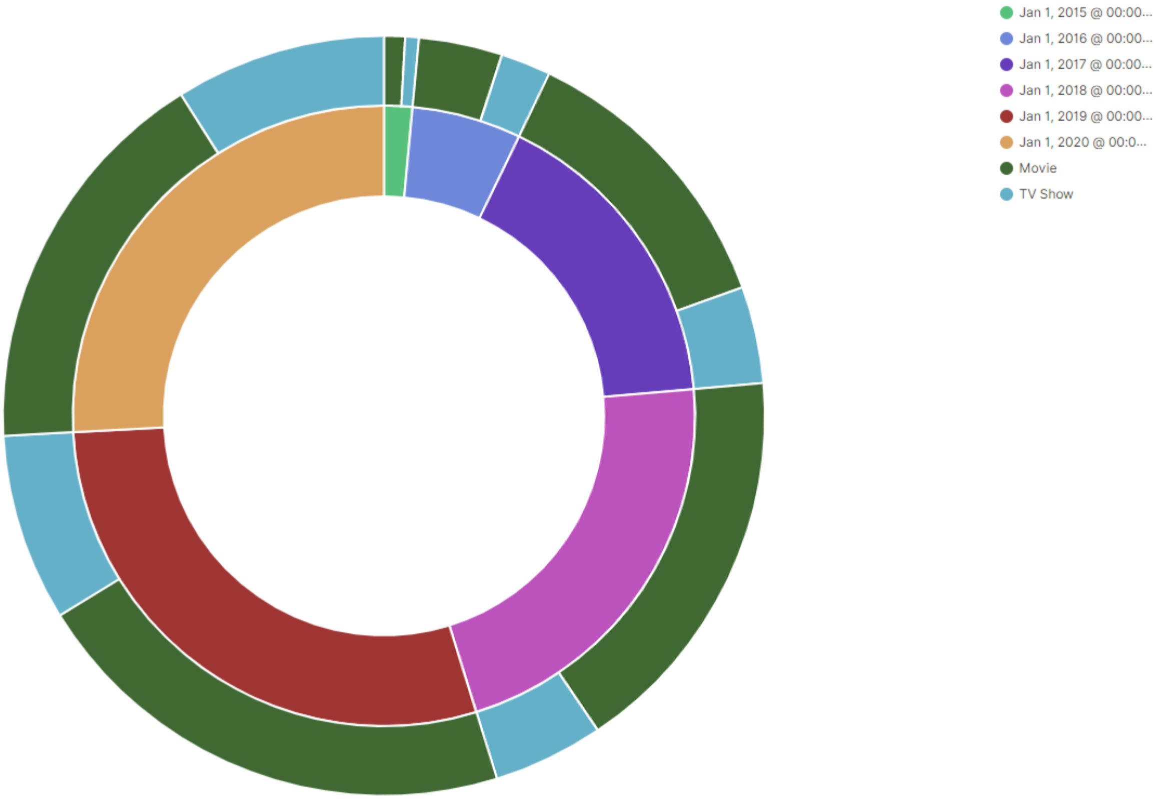
Order

Descending

Size

5

9. Once this is done your pie chart should look like this.



10. Hit save at the top right and save it as **netflixpie**.

Step 6: Creating a Geo-Map

1. Click on the 3 lines in the top left corner to bring up the side bar.

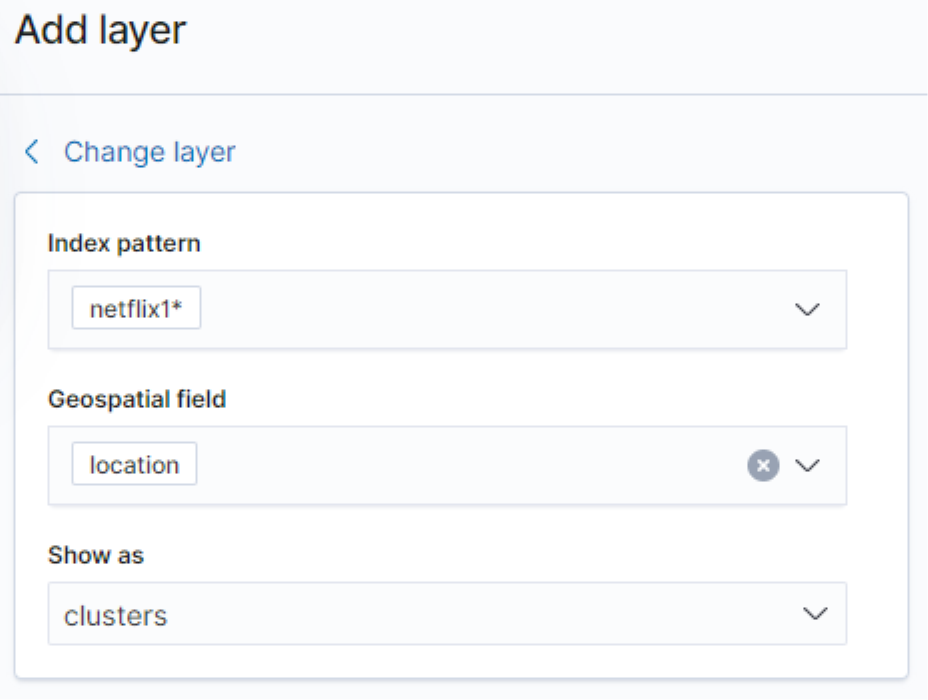
2. Click on **Maps** under the Kibana section.

3. Click on **Create map.**

4. Ensure the dates on the top right matches the previous dates**.**

5. Select add layer and clusters and grids and select the index pattern you previously created.

6. Make sure the layer looks like this and then click add layer.



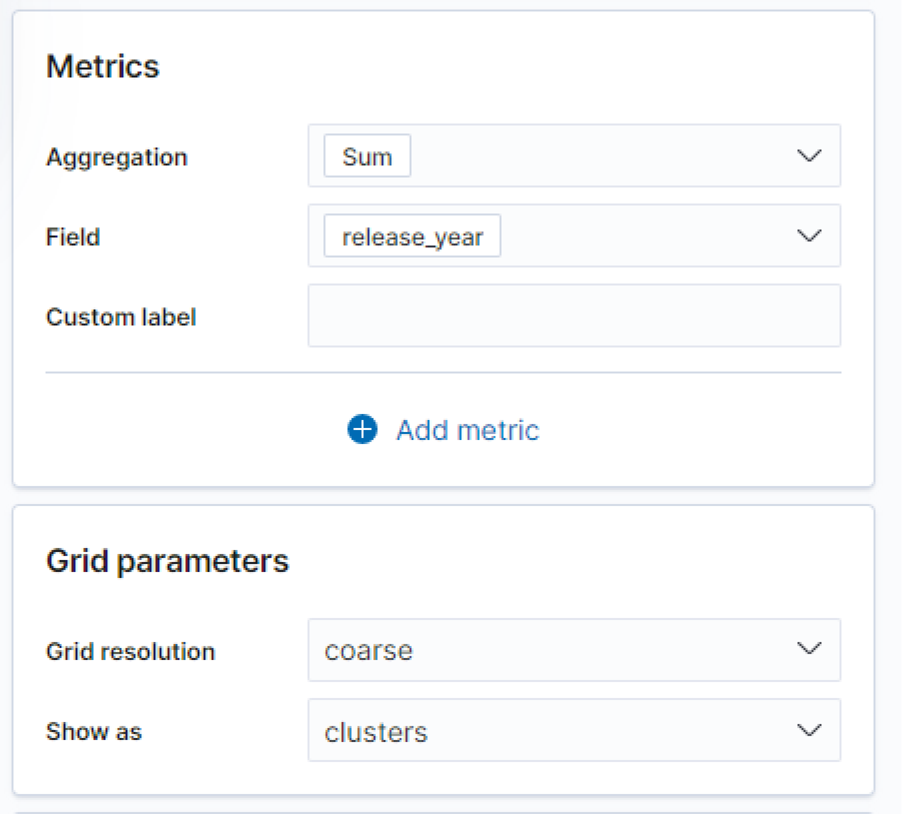
Add layer

Index pattern: netflix1*

Geospatial field: location

Show as: clusters

7. Match the source details for the layer as so.



Metrics

Aggregation: Sum

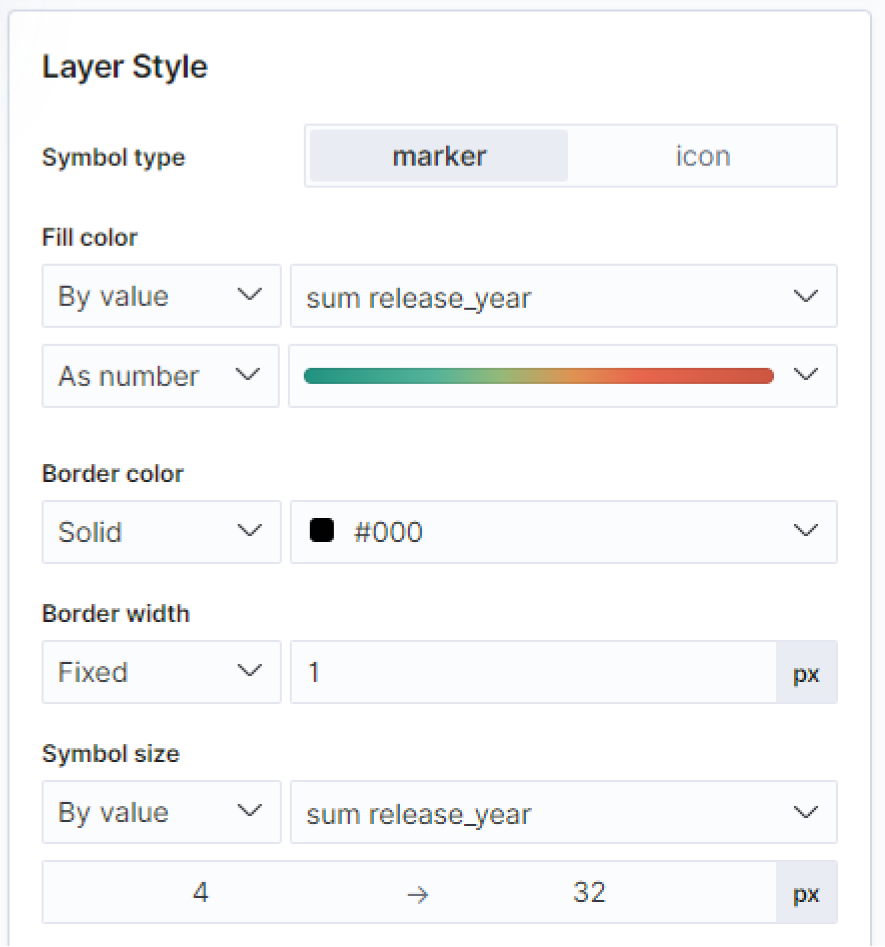
Field: release_year

Custom label:

Grid parameters

Grid resolution: coarse

Show as: clusters



Layer Style

Symbol type: marker | icon

Fill color: By value — sum release_year

As number —

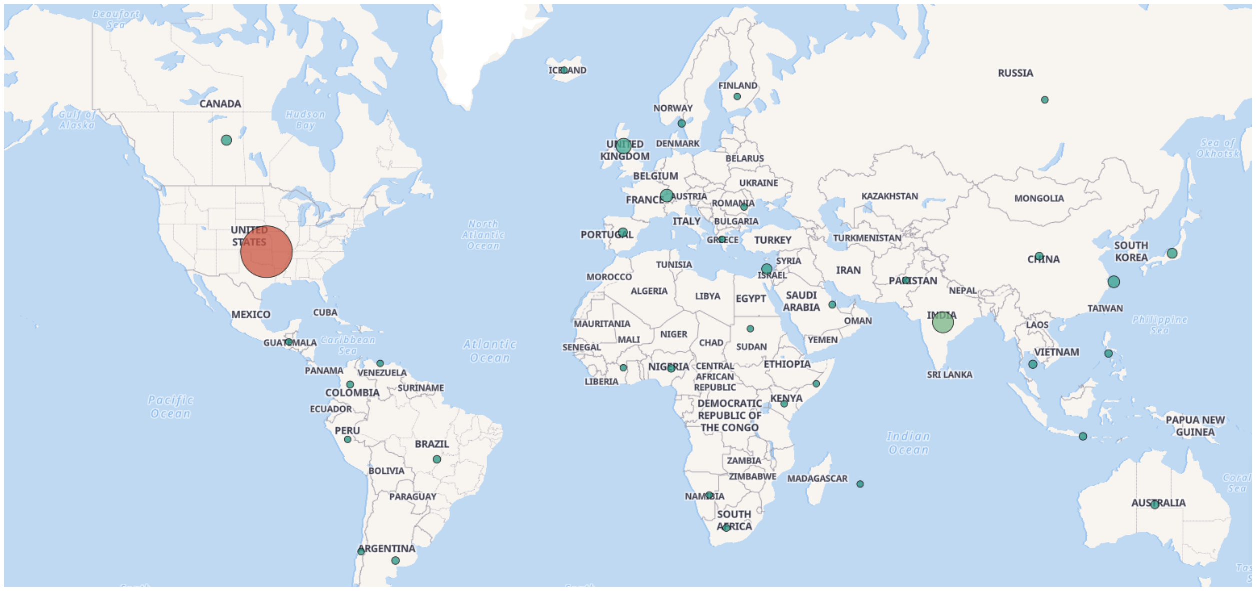
Border color: Solid — #000

Border width: Fixed — 1 px

Symbol size: By value — sum release_year

4 → 32 px

8. Click save and close and save the map as **netflixmap**. It should look similar to this.

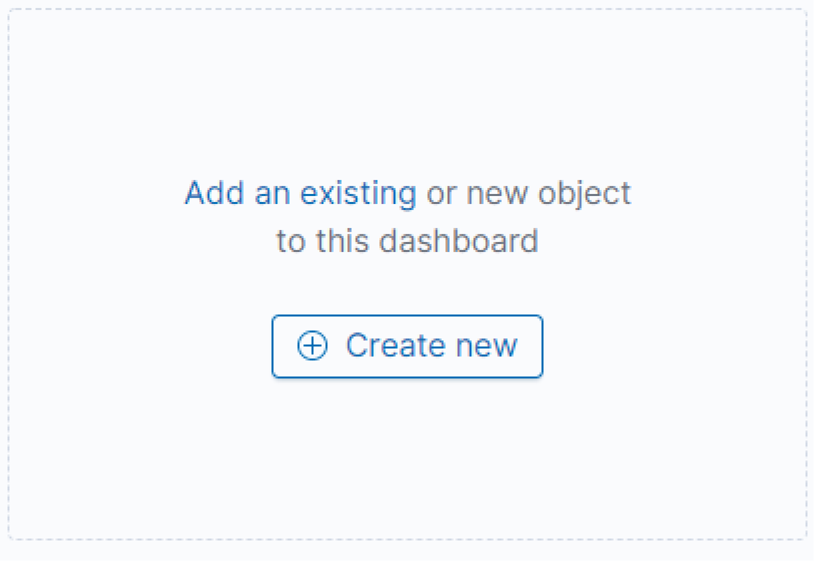


Step 7: Creating a Dashboard

1. Select **Dashboard** under the Kibana section on the left panel.

2. Click on create dashboard and ensure the top right dates match the previous visualizations.

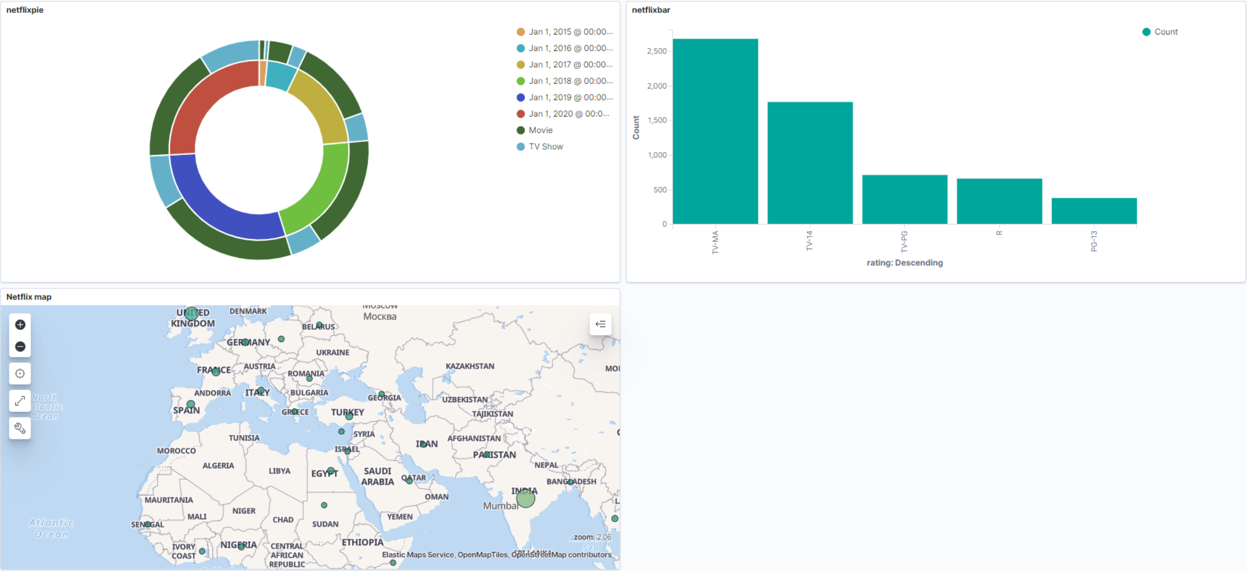
3. Then click **add an existing** in the blank area.



Add an existing or new object to this dashboard

4. Search for netflixbar, netflixpie, and netflixmap and select them.

5. The output should look like this.



6. Hit save at the top right of the interface and save it as **netflixdashboard**.

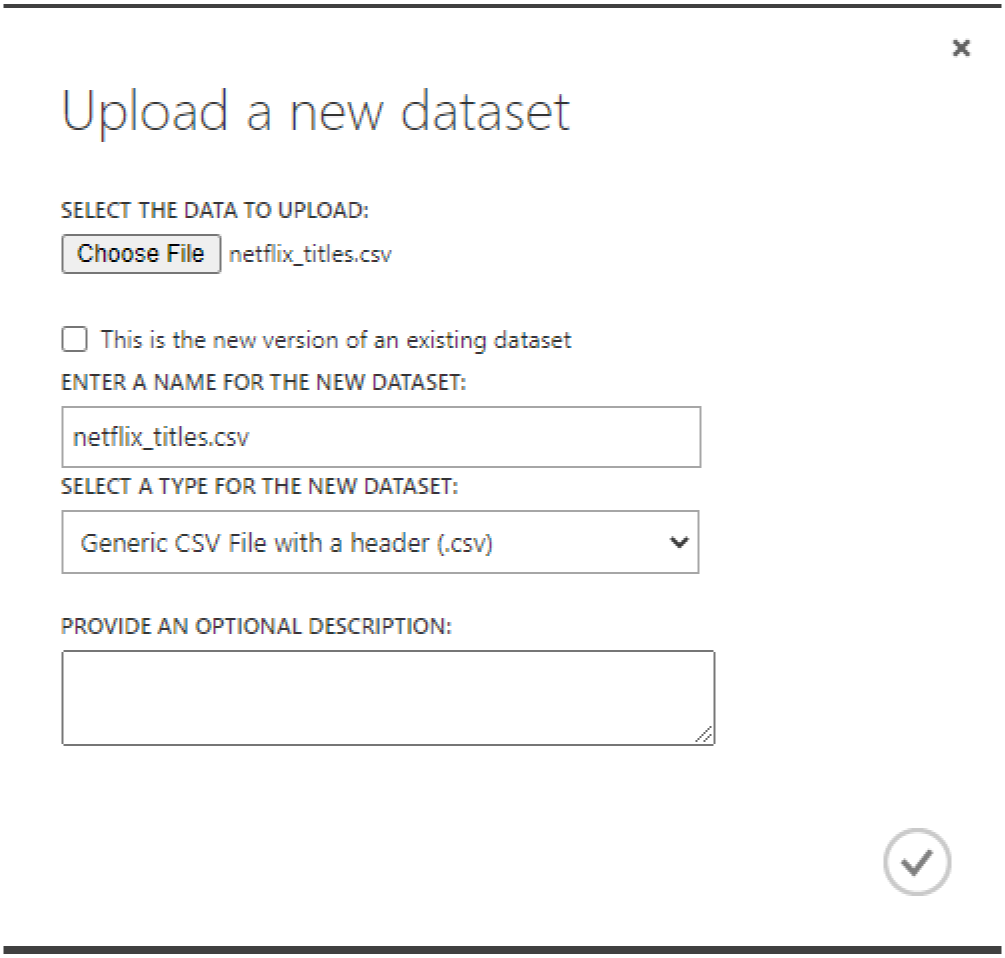
Step 8: Creating AzureML Experiments

Experiment 1:

1. Create an AzureML account at studio.azureml.net, once you log in go to my experiments

2. Once here, hit new at the bottom left to add the dataset

3. Select dataset on the left panel and select local file. Then choose the netflix titles csv from your local machine.



×

Upload a new dataset

SELECT THE DATA TO UPLOAD:

Choose File netflix_titles.csv

□ This is the new version of an existing dataset

ENTER A NAME FOR THE NEW DATASET:

netflix_titles.csv

SELECT A TYPE FOR THE NEW DATASET:

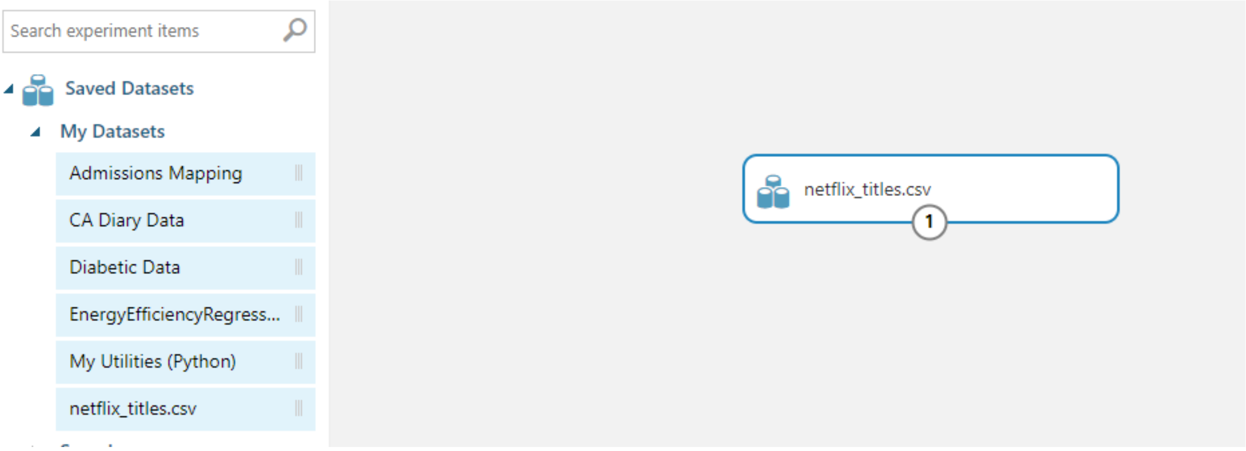
Generic CSV File with a header (.csv)

PROVIDE AN OPTIONAL DESCRIPTION:

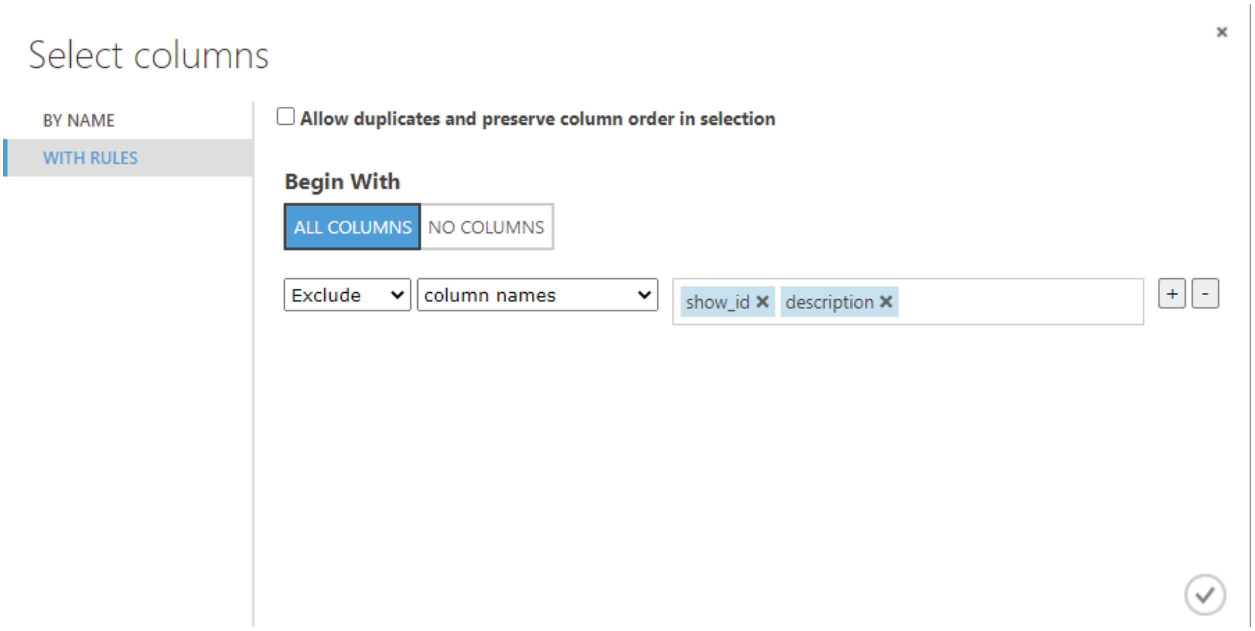
✓

Click the checkmark button and wait for the csv to upload.

4. Once it has finished click the new button once again and create a blank experiment. Rename the experiment Netflix titles and drag the dataset onto the canvas.



5. Search for the select columns in dataset module and drag it under the dataset. Then connect the output of netflix_titles to the input of the select columns module.
6. Select the Select columns in dataset module and go to the properties pane, launch the column selector and configure it like this.



Select columns

BY NAME

WITH RULES

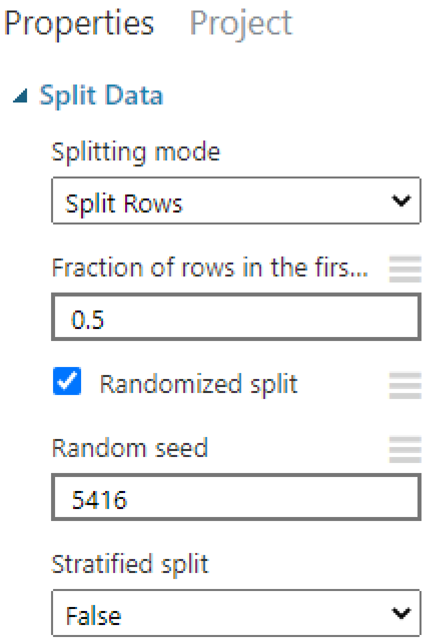
□ Allow duplicates and preserve column order in selection

Begin With

ALL COLUMNS | NO COLUMNS

Exclude | column names | show_id ✕ description ✕

7. Then search for the split data module and drag it below the select columns in dataset module. Configure it as so.



Properties Project

Split Data

Splitting mode

Split Rows

Fraction of rows in the firs...

0.5

☑ Randomized split

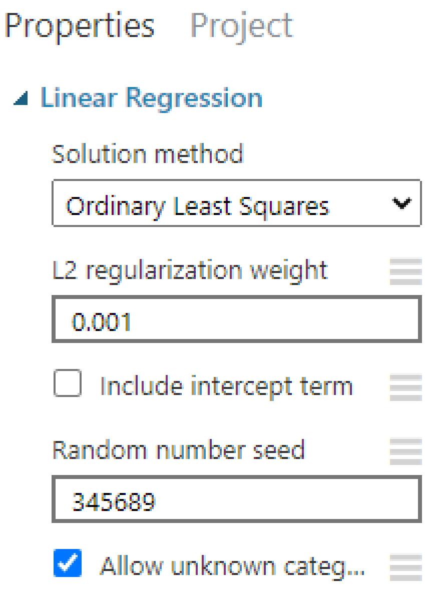
Random seed

5416

Stratified split

False

8. Search for the linear regression module and place it on the side of the split data module. Configure it like this.



Properties Project

Linear Regression

Solution method

Ordinary Least Squares

L2 regularization weight

0.001

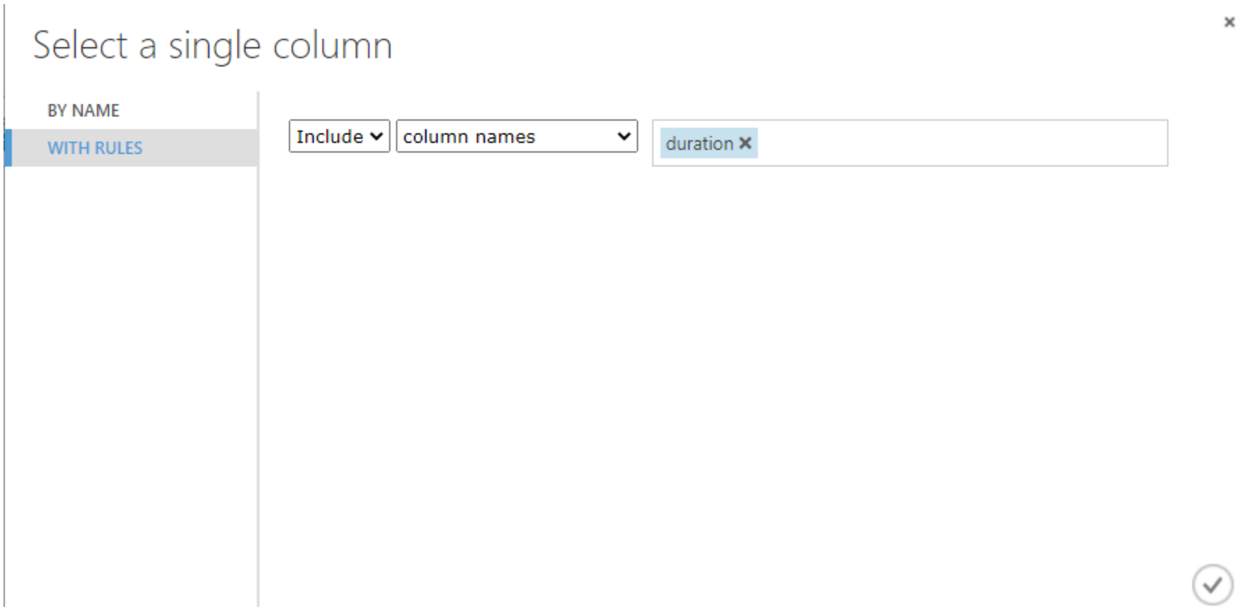
□ Include intercept term

Random number seed

345689

☑ Allow unknown categ...

9. Search for the Train model module and place it under the linear regression and split data modules. Select the train model module and launch the column selector and select duration.



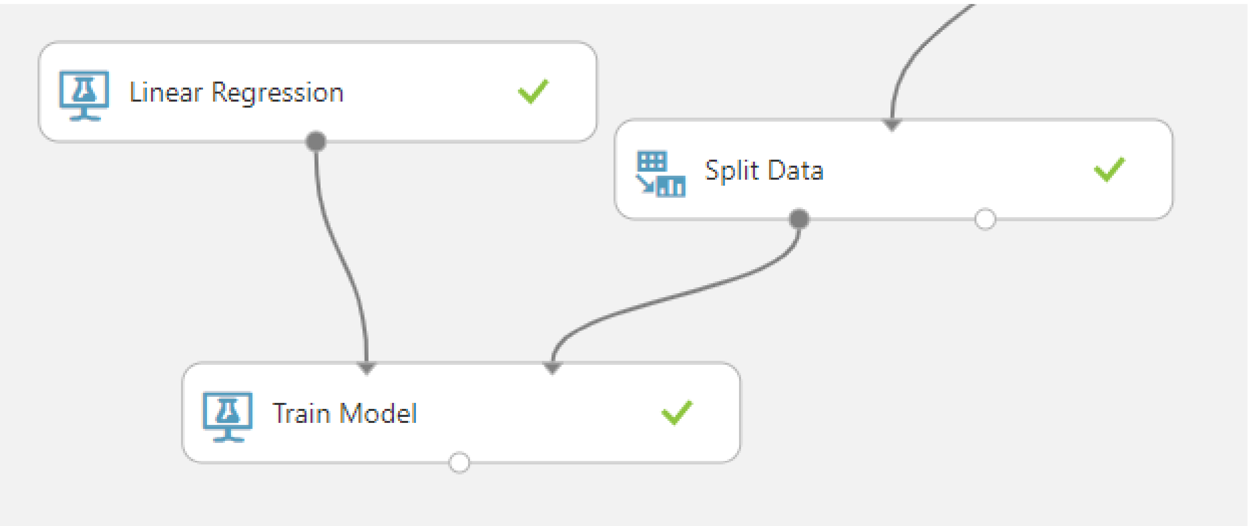
Select a single column

BY NAME

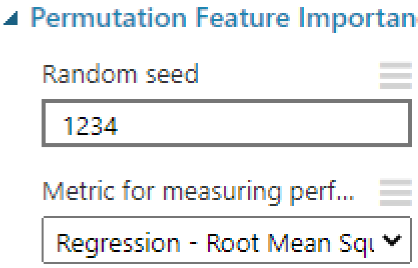
WITH RULES

Include | column names | duration ✕

10. Connect the output of linear regression to the left input of train model and the left output of split data to the right input of train model as seen below.



11.Search for the Permutation Feature Importance module and place it under the Train Model module with these properties.



◢ Permutation Feature Importance

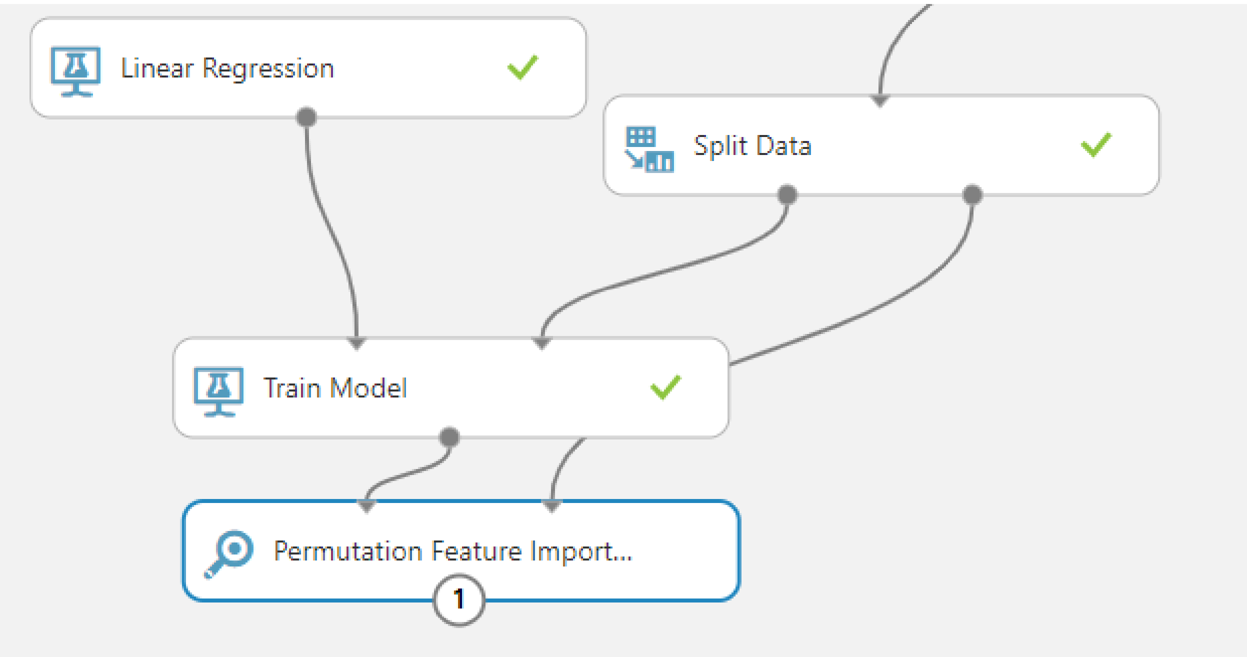
Random seed

1234

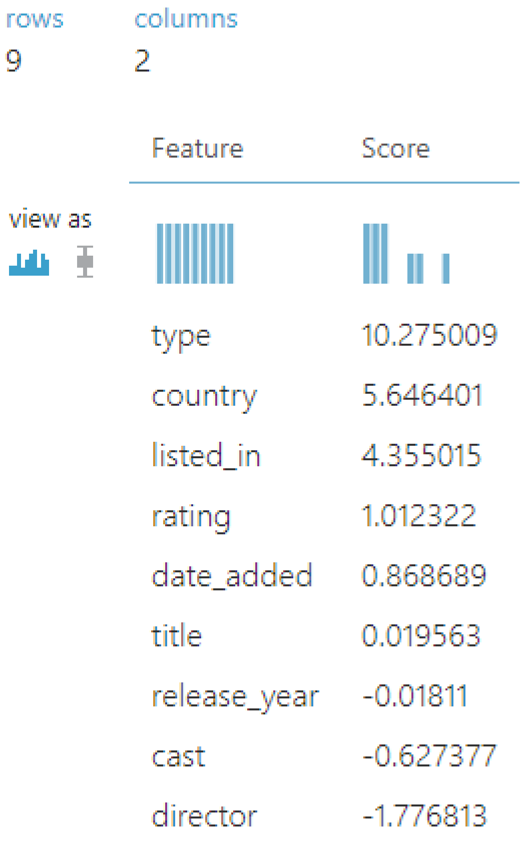
Metric for measuring perf...

Regression - Root Mean Sqı

12. Connect the output of train model to the left input of permutation feature importance and connect the right output of split data to the right input as seen below. Then save and run the experiment.



13. After running the experiment, visualize the permutation feature importance, it looks like this.



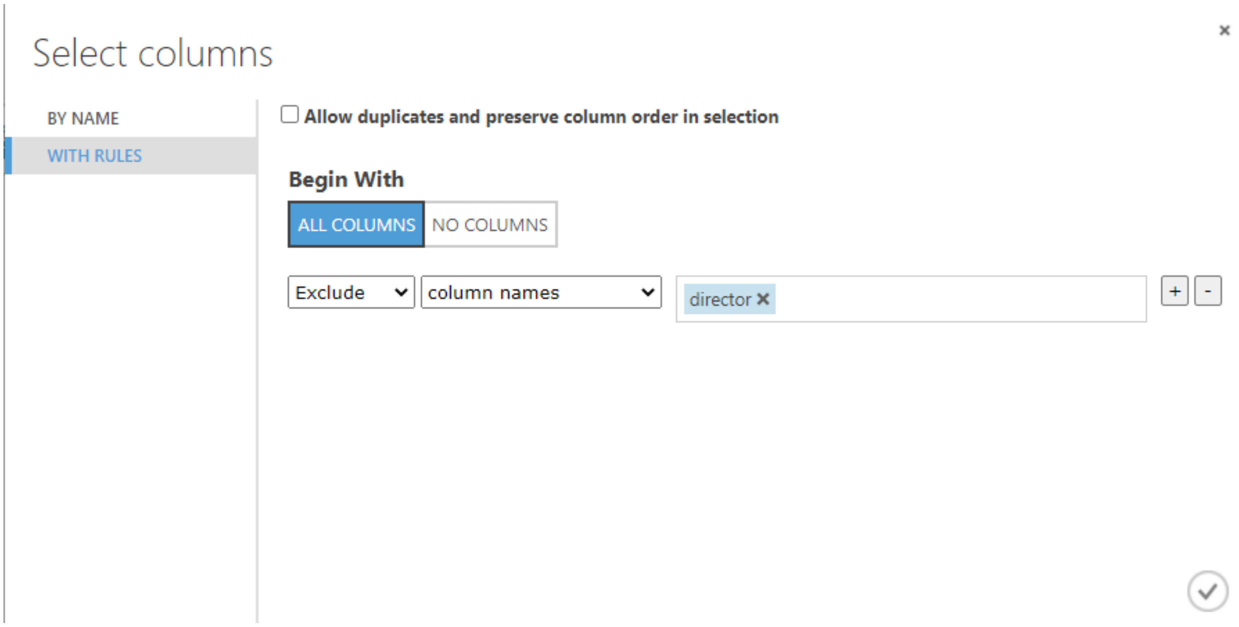
rows 9

columns 2

view as

Feature	Score
type	10.275009
country	5.646401
listed_in	4.355015
rating	1.012322
date_added	0.868689
title	0.019563
release_year	-0.01811
cast	-0.627377
director	-1.776813

14. Add another Select columns in dataset module under the first and configure it like this.



Select columns

BY NAME

WITH RULES

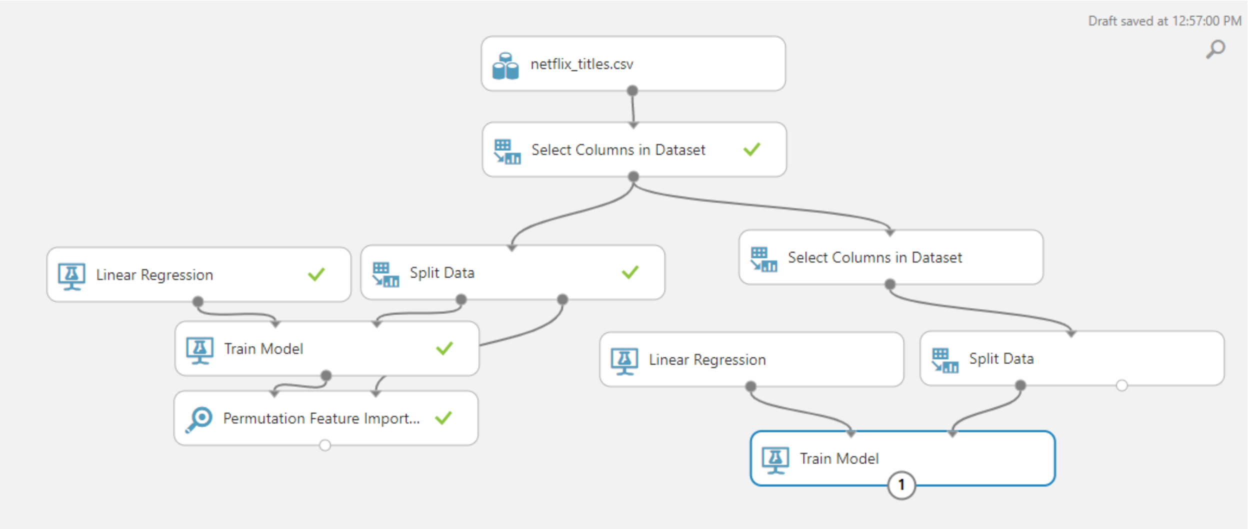
☐ Allow duplicates and preserve column order in selection

Begin With

ALL COLUMNS | NO COLUMNS

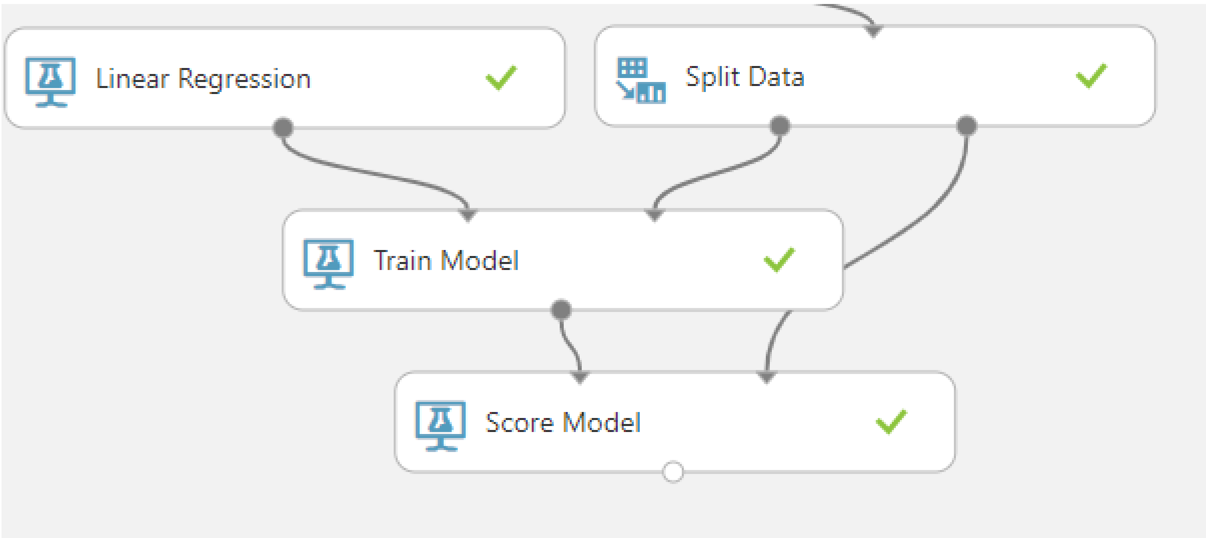
Exclude | column names | director ✕

15. Click and drag around the Linear regression, split data, and train model modules and copy paste them under the new select columns module. Then connect the output of the new select columns in dataset to the new split data input. The experiment should look like this.

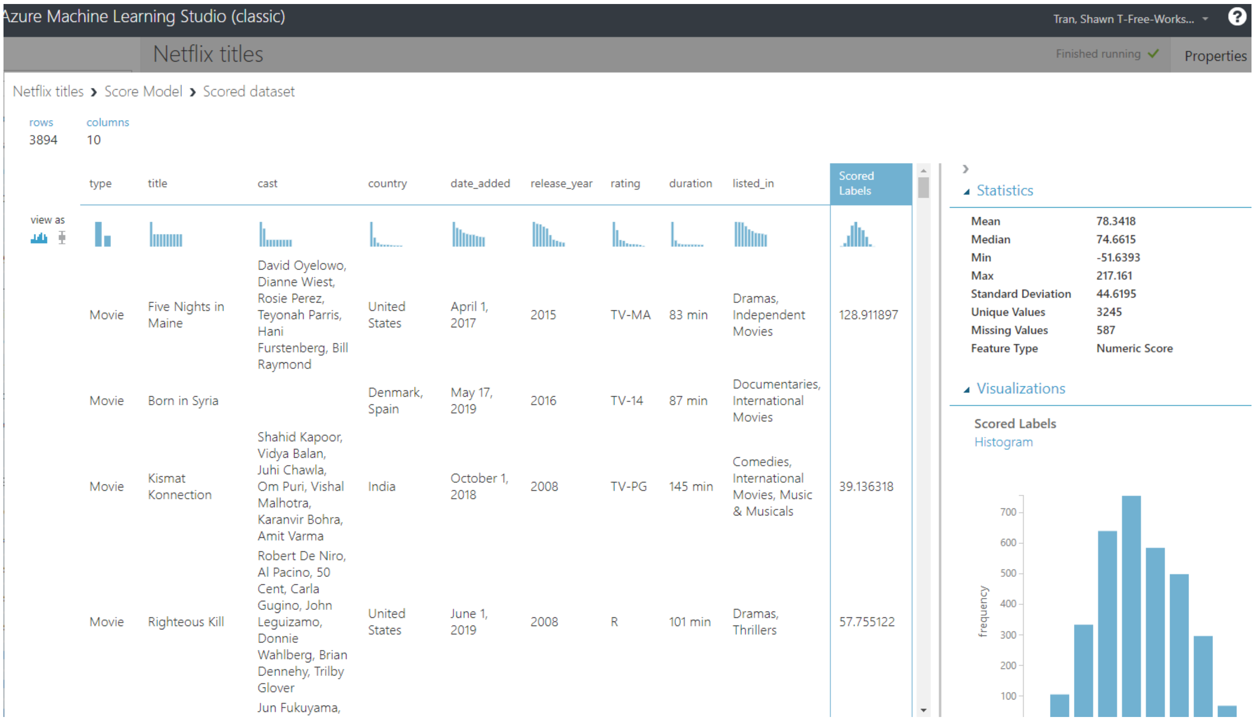


Draft saved at 12:57:00 PM
netflix_titles.csv
Select Columns in Dataset ✓
Linear Regression ✓
Split Data ✓
Select Columns in Dataset
Train Model ✓
Linear Regression
Split Data
Permutation Feature Import... ✓
Train Model
1

16. Search for the Score model module and add it under the train model module. Connect the output of the train model to the left input of score model and the right output of split data to the right input of score model as seen below.



17. Save and run the entire experiment and visualize the scored dataset, it should look like this.



rows 3894

columns 10

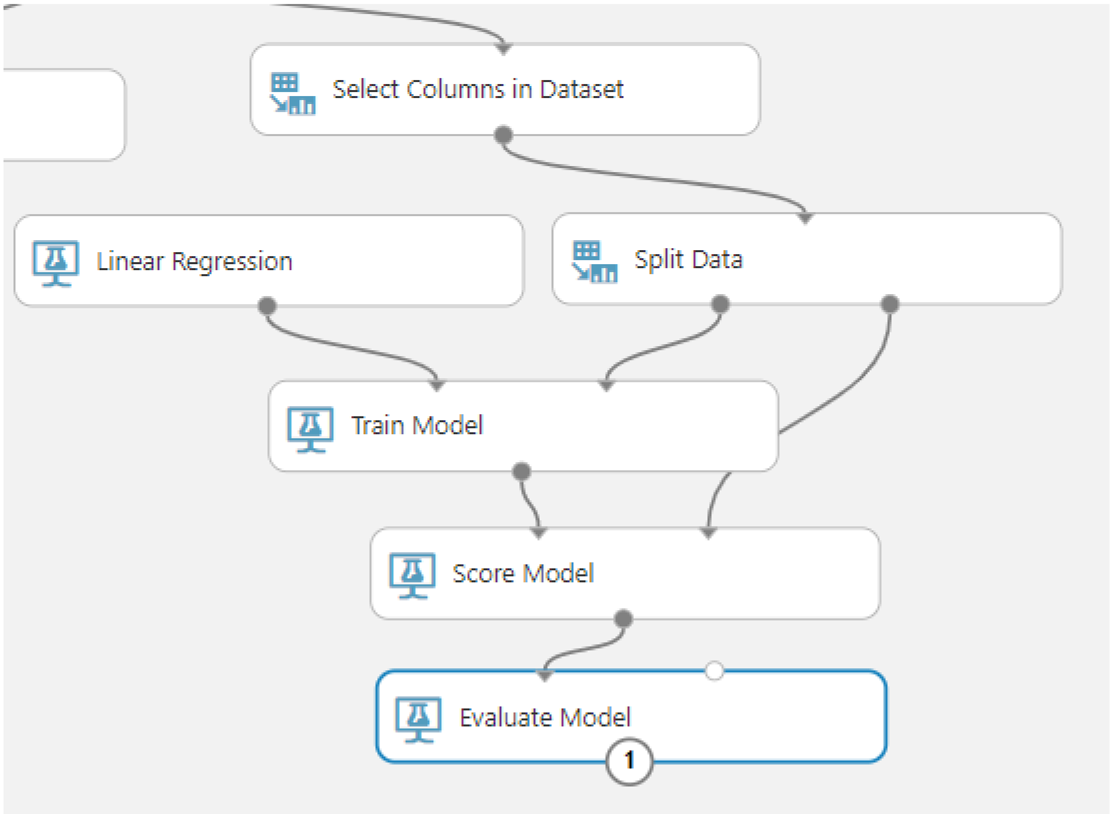
type	title	cast	country	date_added	release_year	rating	duration	listed_in	Scored Labels
Movie	Five Nights in Maine	David Oyelowo, Dianne Wiest, Rosie Perez, Teyonah Parris, Hani Furstenberg, Bill Raymond	United States	April 1, 2017	2015	TV-MA	83 min	Dramas, Independent Movies	128.911897
Movie	Born in Syria		Denmark, Spain	May 17, 2019	2016	TV-14	87 min	Documentaries, International Movies	
Movie	Kismat Konnection	Shahid Kapoor, Vidya Balan, Juhi Chawla, Om Puri, Vishal Malhotra, Karanvir Bohra, Amit Varma	India	October 1, 2018	2008	TV-PG	145 min	Comedies, International Movies, Music & Musicals	39.136318
Movie	Righteous Kill	Robert De Niro, Al Pacino, 50 Cent, Carla Gugino, John Leguizamo, Donnie Wahlberg, Brian Dennehy, Trilby Glover	United States	June 1, 2019	2008	R	101 min	Dramas, Thrillers	57.755122
		Jun Fukuyama,							

Statistics

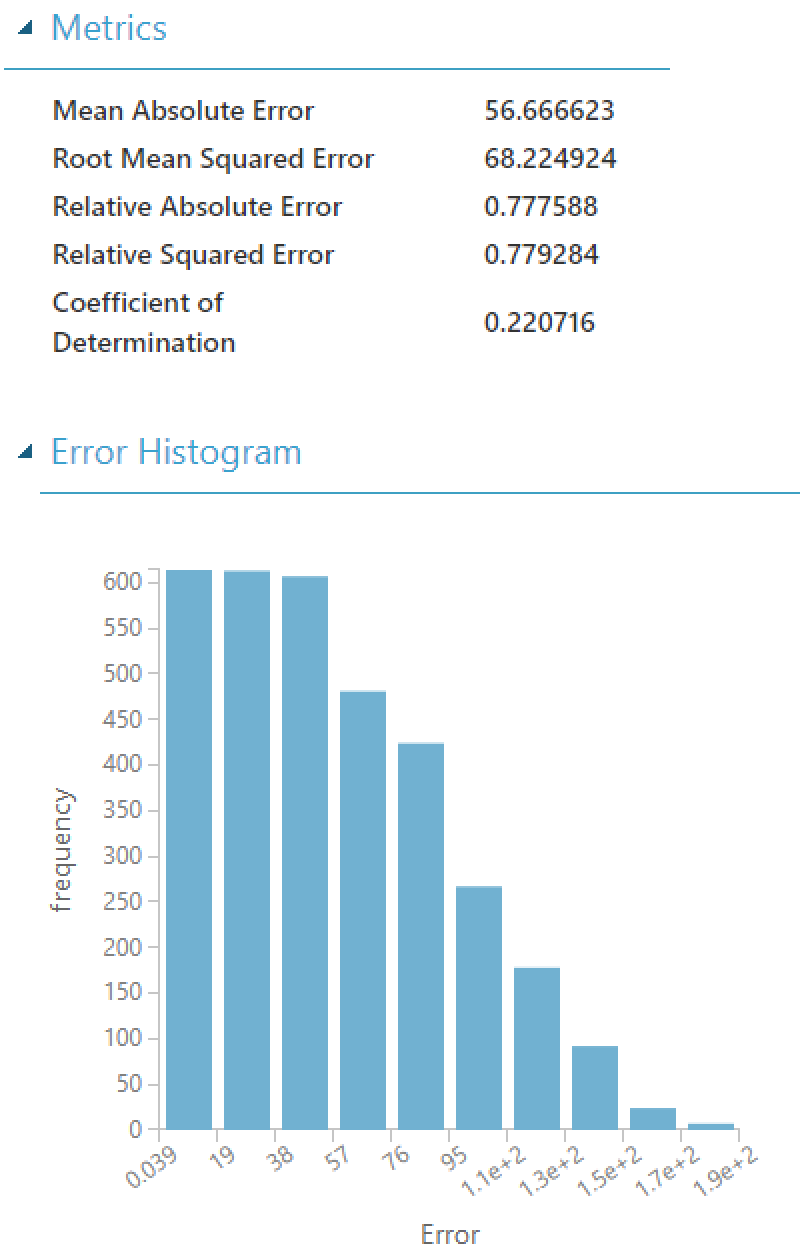
Mean	78.3418
Median	74.6615
Min	-51.6393
Max	217.161
Standard Deviation	44.6195
Unique Values	3245
Missing Values	587
Feature Type	Numeric Score

Experiment 2:

1. Go to studio.azureml.net and load the previous experiment netflix titles. Save it as a copy named netflix titles 2.
2. Search for Evaluate model and drag it under the right side set of modules like so.



3. Save and run the experiment and visualize the output of the evaluate model module.

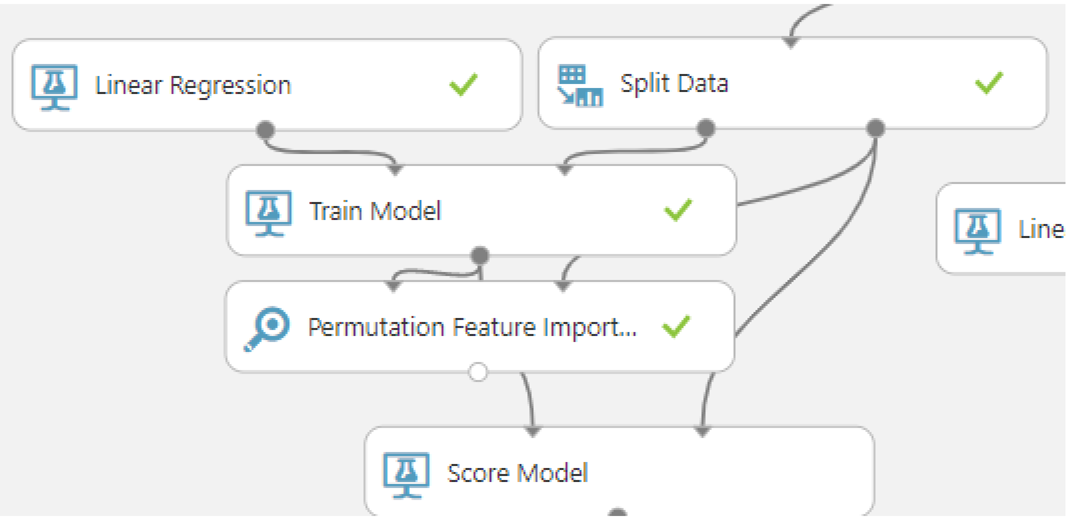


◢ Metrics

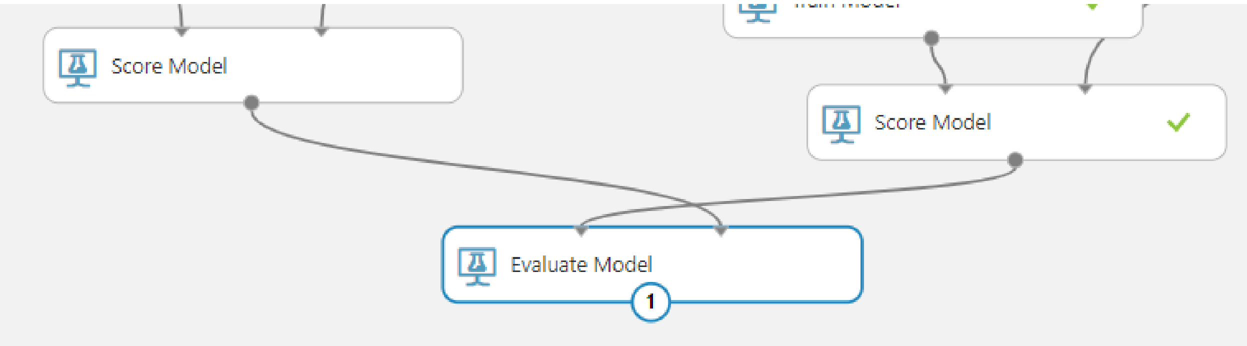
Mean Absolute Error	56.666623
Root Mean Squared Error	68.224924
Relative Absolute Error	0.777588
Relative Squared Error	0.779284
Coefficient of Determination	0.220716

◢ Error Histogram

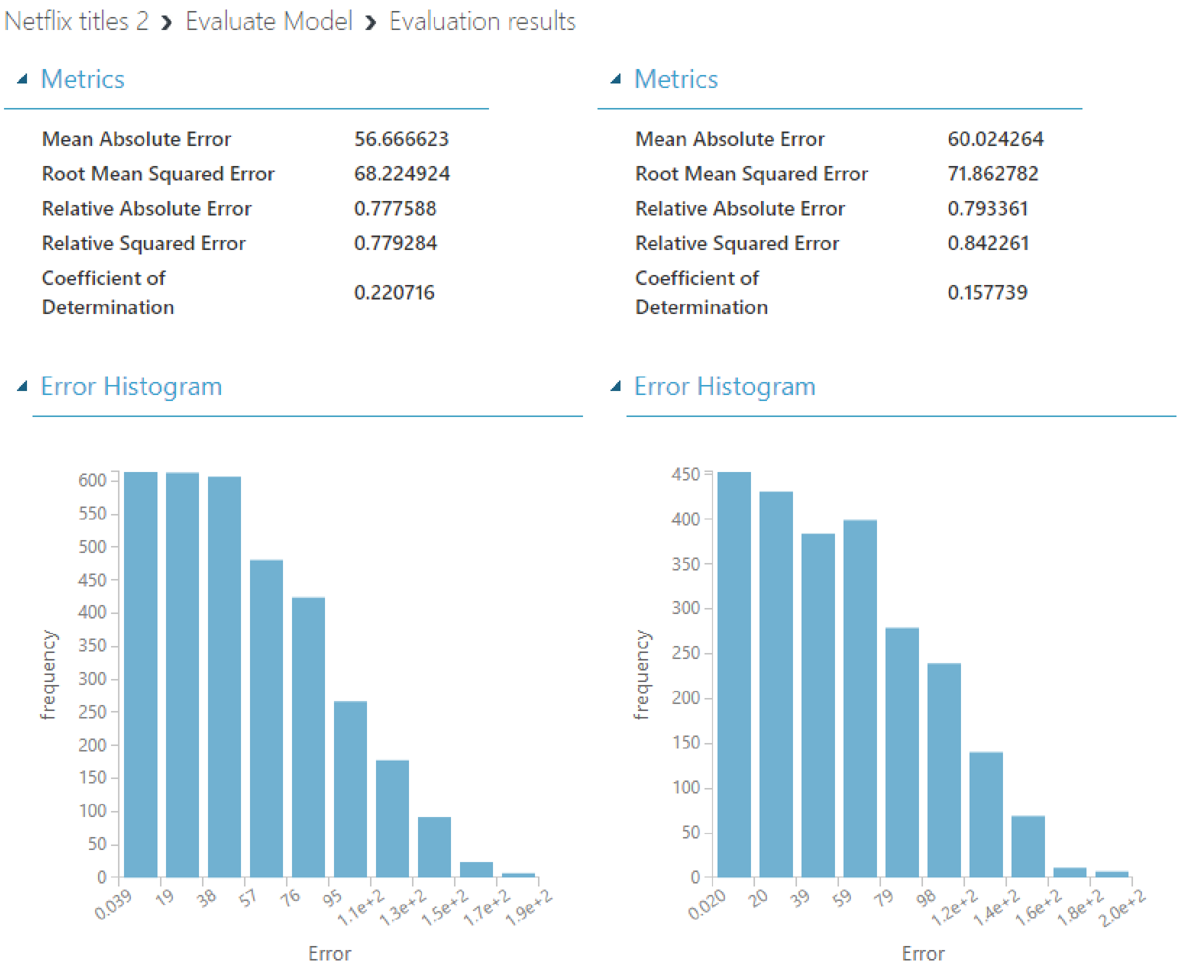
4. Search for the score model module and place it under the left set of modules under permutation feature importance. Connect the train model output to the left input of the new score model module and connect the right output of the split data module to the right input of the new score model module like so.



5. Then connect the output of the new score model to the right input of the evaluate model as seen below. Then save and run the experiment.



6. Visualize the results of the evaluate model module, it looks like this.



Netflix titles 2 › Evaluate Model › Evaluation results

◢ Metrics

Mean Absolute Error	56.666623
Root Mean Squared Error	68.224924
Relative Absolute Error	0.777588
Relative Squared Error	0.779284
Coefficient of Determination	0.220716

◢ Metrics

Mean Absolute Error	60.024264
Root Mean Squared Error	71.862782
Relative Absolute Error	0.793361
Relative Squared Error	0.842261
Coefficient of Determination	0.157739

◢ Error Histogram

◢ Error Histogram

Here we see the first model we evaluated on the left has a better performance, but both models are still not ideal as the RMSE are both above 65 and the CoD are both under 25.

This is the end of the lab.