
Visualize data with QuickSight

Sreeramjvp



Amazon QuickSight.

What it does & how it's useful

.....

Amazon QuickSight.

Developers and teams use Amazon QuickSight because it is easy to use, scalable, and offers robust data visualization capabilities. It allows users to create interactive dashboards, gain insights from their data quickly, and make data-driven decisions without needing extensive technical expertise.

How I'm using it in today's project

.....

I'm using Amazon QuickSight in this project to create interactive dashboards and visualizations that provide real-time insights into our data. By leveraging QuickSight's powerful analytics capabilities, we can make data-driven decisions more efficiently and effectively, ultimately driving the success of our project.

This project took me...

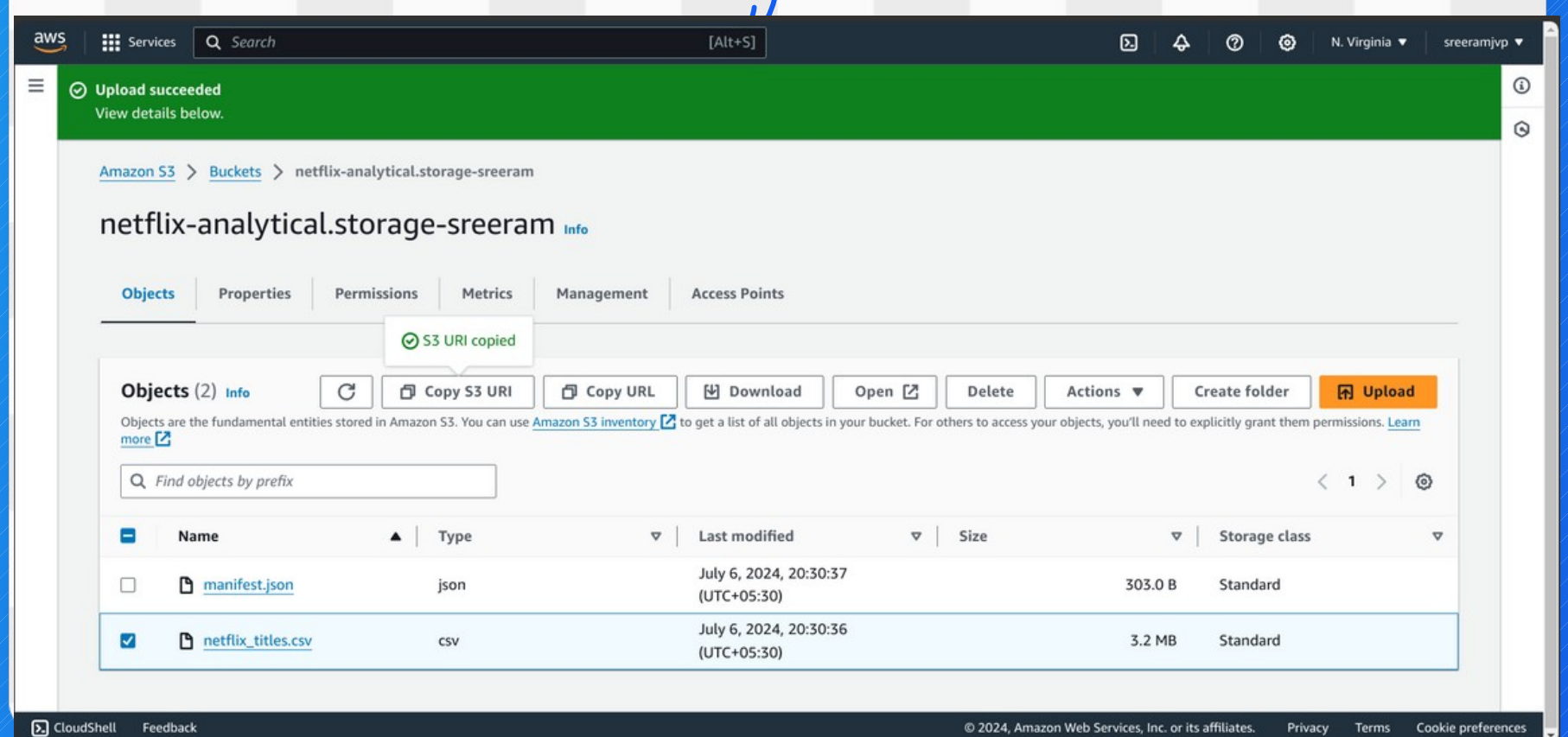
[how much time did this project take you to complete?]

Documentation took me around two weeks to complete, given the extensive research and detailed analysis required. Writing the documentation itself took about one week, ensuring clarity and comprehensiveness. Overall, the process was thorough and rewarding.

Upload project files into S3

- S3 is used in this project to store two files, which are:
 1. A .csv file that contains all of the value metrics of all the shows, ratings, user perception, trend analysis.
 2. A .json file, which consists of information that relays the s3 bucket address and a plethora of web access information required and exchanged between various AWS entities.
- I edited the manifest.json file by using an inbuilt text-editor, packaged within the linux system. The .json points towards the location of s3 bucket, where the .csv file is present.

Here's my bucket with the CSV file and manifest.json!



Create QuickSight account

- It costed absolutely nothing for the requirements I needed.
- Took care of proper checks, to ensure this wont incur any charges.
- Creating a QuickSight account took me nearly 20 minutes of time. The time included going through the documentation of the quicksight service, properly assessing the requirements and managing the billing.
- I also had to enable QuickSight's access to S3 because, it is required to choose the specific s3 bucket that would be required for quicksight report.

Voila! I created my QuickSight account successfully.



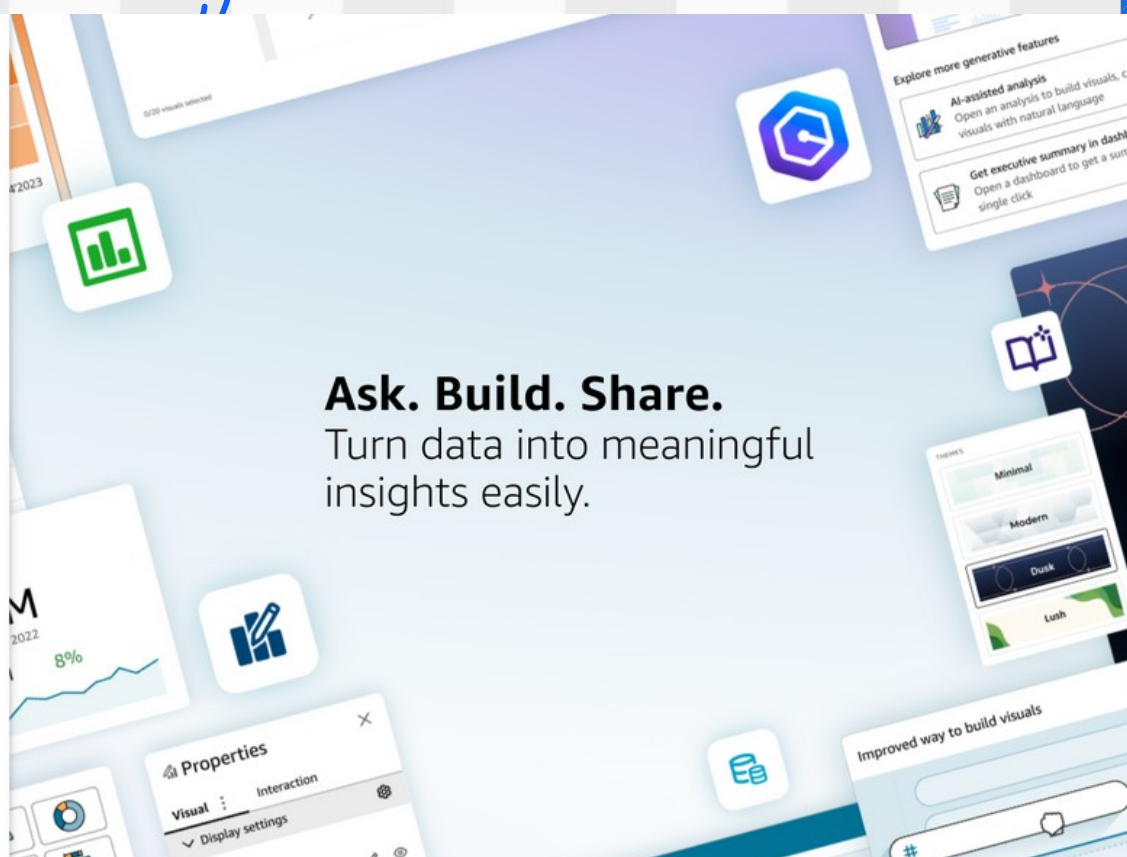
Account created successfully

Account name: sreeramjvp-analytics

[GO TO QUICKSIGHT](#)

Ask. Build. Share.

Turn data into meaningful insights easily.



Connect S3 + QuickSight

- I connected the S3 bucket to QuickSight by manually choosing the s3 bucket I have created, and uploading the manifest.json, which serves as a map to Quicksight to search and look for data.
- The manifest.json file was important in this step because manifest.json tells QuickSight what the dataset looks like, so QuickSight knows how to understand the data and show it in charts or graphs. Without this map, QuickSight might get confused and not show your data correctly! manifest.json tells QuickSight what the dataset looks like, so QuickSight knows how to understand the data and show it in charts or graphs.

Entering the manifest.json URL.

New S3 data source ×

Data source name

kaggle-netflix-data

Upload a **manifest file**

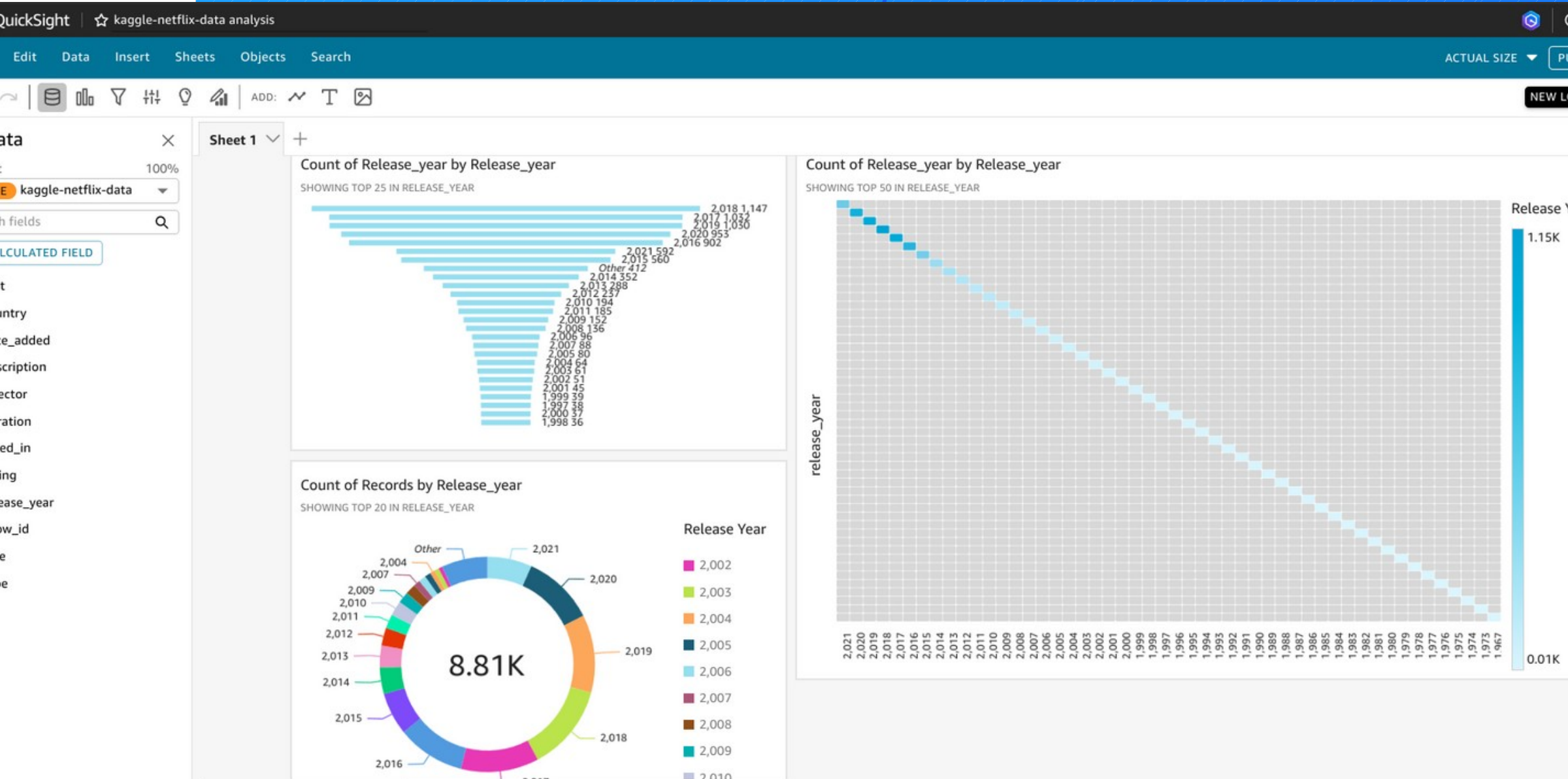
☒ URL ☐ Upload

s3://netflix-analytical.storage-sreeram/manifest.json

Connect

Let's make visualisations!

- To create visualization on QuickSight, you'll have to select the recognized attributes and drag them onto the canvas. After choosing your attribute, we can select from the many of the list of visualization graphs.
- The chart/graph shown here is a breakdown of the counts of movies and shows released over a period of an year.
- I created this graph by choosing the desired attribute and applying the several visualize tools available.

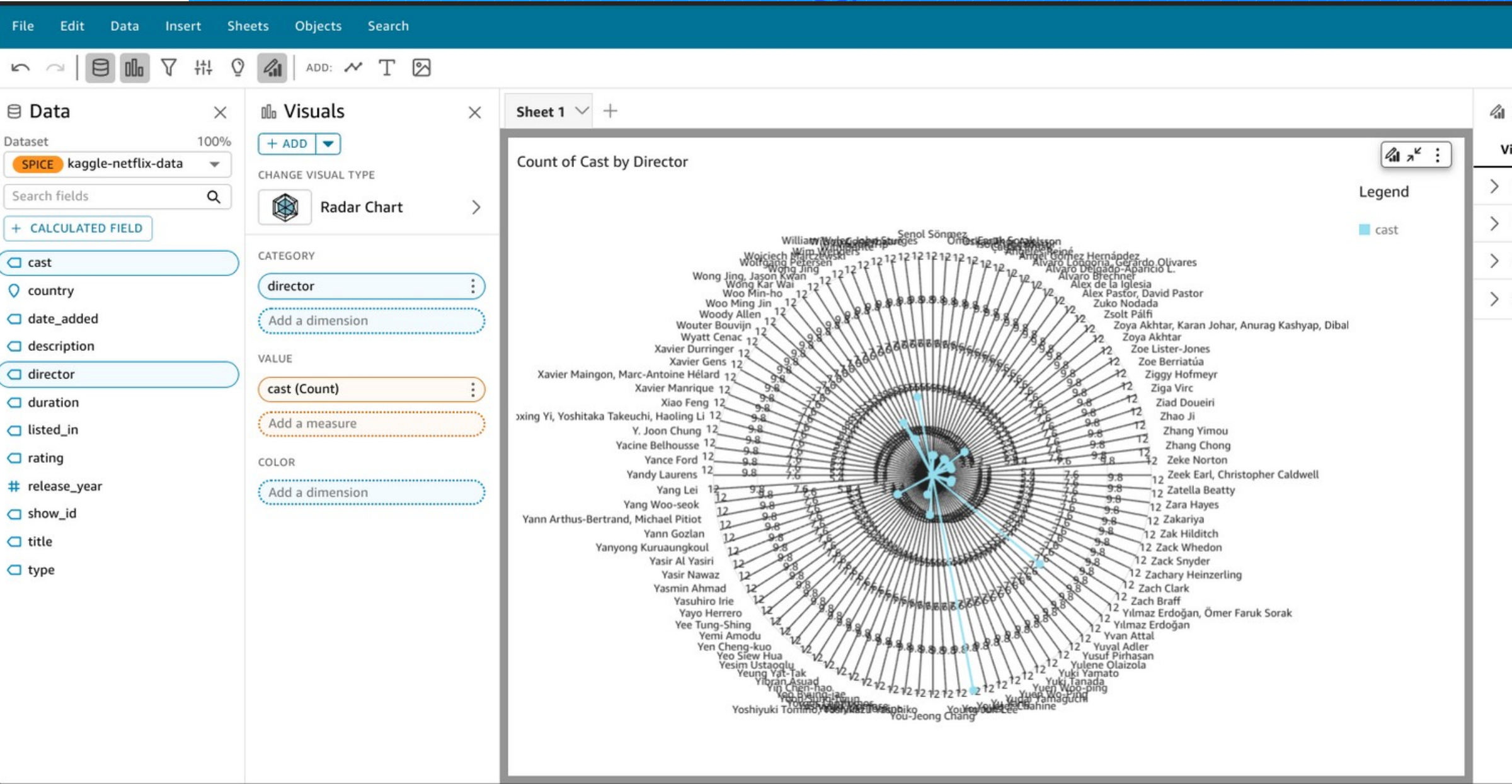


sreeram

Using filters

- Filters are useful for mixing and combining several aspects of the data. They are useful to compare several attributes against each other for better insights into the data.
- Here I added a filter by using tags and dragging them into their fields. This helped me create a visualisation on how each metric is co-related to each other and their influence on other metrics.

A visualisation set up after adding filters.



Setting up my dashboard..

- The final dashboard can be seen here, with a wide array of visualization graphs, with filters applied.

Voila! Here's the finished dashboard!



My key learnings

- 1** An S3 bucket was used in this project to store objects. Objects can be in any form, and can be uploaded to s3 using: AWS console, AWS CLI, sdk's etc..
- 2** To connect the data stored in S3 with QuickSight, I had to use a manifest.json file which acts as a map for interconnection between different services within AWS.
- 3** Creating visualization with quicksight is a breeze, with a suite of inbuilt tools that made adding, applying, filtering and categorizing the data very easy.
- 4** One thing I didn't expect was the level of fine-tuning that's possible within the aws service to be able to adjust and tune the function metrics to our liking. The minute details with the services and the level of attention given towards how resources are handled, control and flexibility aws provides to make the user experience as seamless as possible.