EXPLORING TWO OPEN SOURCE BI TOOLS: APACHE SUPERSET & METABASE

A PROJECT REPORT

Submitted for Data Science Internship at ISI, Kolkata

BACHELOR OF SCIENCE
IN
ECONOMICS

by

MITHI DEY

Under the guidance of

MONALISA PAL

IDEAS - Institute of Data Engineering, Analytics and Science Foundation Technology Innovation Hub @ Indian Statistical Institute, Kolkata
203 Barrackpore Trunk Road, Kolkata-700108, INDIA

October 2024

Acknowledgment

I would like to express my thanks and acknowledgement to **Agnimitra Biswas**, the CEO of IDEAS- Institute of Data Engineering, Analytics and Science Foundation Technology Innovation Hub @ Indian Statistical Institute, Kolkata, for giving me the opportunity to work on this project.

Specially, I would like to express my thanks and gratitude to my project supervisor **Monalisa Pal**, for her immense support, guidance, and valuable advice to complete this project, titled as – **"Exploring Two Open Source BI Tools : Apache Superset & Metabase"**. This project has helped me to explore so many things related to the topic. It has been a great experience to work under the supervision of **Monalisa Pal**, whose valuable lessons and suggestions have really enriched the content of my project work.

Contents

1 Project Overview **4**

2 Work Approach **4**

3 Dataset Overview **7**

4 Installation Steps of Apache Superset & Metabase **9**
4.1 Installing Superset using Docker Compose 9
4.2 Installing Metabase using Docker Compose 10

5 Step-by-Step Guide to Data Import **11**
5.1 Connecting MySQL Database to Apache Superset 11
5.2 Connecting MySQL Database to Metabase 12

6 Dashboard **13**
6.1 Dashboard Creation in Apache Superset: 14
6.2 Example : Building an Agricultural Production Dashboard in Apache Superset 15
6.3 Dashboard Creation in Metabase: 25
6.4 Example : Building an Agricultural Production Dashboard in Metabase . 25

7 Embedding Superset & Metabase dashboard into HTML **32**
7.1 Steps to Embed Superset Dashboard into HTML 34
7.2 Steps to Embed Metabase Dashboard into HTML 36

8 Apache Superset vs Metabase **38**
8.1 Data Sources 38
8.2 No-code Query Builder 38
8.3 Visualization Features 38
8.4 Advanced SQL Editor 39

9 Conclusion and Recommendation **40**

10 Future Exploration of Apache Superset and Metabase **41**

11 Appendix **43**

1 Project Overview

Data visualization and BI tools are essential for any modern organization to make fast, data-driven decisions. With a wide range of visualization tools available to choose from, it's important to know what makes each platform unique. Most BI platforms allows to create data visualizations, perform complicated data analysis, and build interactive dashboards, open source BI tools offer even more benefits. Whether you're looking for cost effectiveness, customizability, wide-spread community support, enhanced security, or future-proofing features, open source BI is the way to go. Apache Superset and Metabase are the two popular open-source business intelligence platforms.

The objective of this project is to evaluate and compare two popular open-source business intelligence (BI) tools, Apache Superset and Metabase, by creating parallel, fully functional dashboards using the same dataset. These will demonstrate the features and capabilities of each tool, allowing for a comprehensive comparison based on visualizations, user interactions, ease of use, and advanced functionalities.

2 Work Approach

1. **Data Collection:** Gathered the dataset in PDF format from the Department of Agriculture and Farmers Welfare, which includes agricultural production records.

2. **Data Conversion:** Converted the dataset from PDF format to a CSV file for compatibility with database software.

3. **Database Setup in SQL Workbench:**

    3.1. Accessed SQL Workbench.

    3.2. Created a schema specific to the project.

    3.3. Imported the CSV file to create a crop production table within this schema.

4. **Connecting Database to Apache Superset:**

    4.1. Configured Apache Superset to connect to the SQL Workbench database.

    4.2. Established connection settings and tested for successful integration.

5. **Dashboard Creation in Apache Superset:**

    5.1. Created various visualizations (e.g., pie charts, bar graphs) using the crop data.

    5.2. Designed a dashboard incorporating these charts.

    5.3. Added filters and customized the layout, title, and other dashboard properties.

6. **Connecting Database to Metabase:**

    6.1. Connected the database to Metabase by configuring the required connection settings.

    6.2. Ensured the data was accessible and visible in Metabase.

7. **Dashboard Creation in Metabase:**

    7.1. Created visualizations and charts using the same dataset in Metabase.

7.2. Built a dashboard, including the charts and filters, and customized its layout and appearance.

7.3. Saved the dashboard for further comparison.

8. **Dashboard Embedding:**

8.1. Implemented dashboard embedding in both Apache Superset and Metabase.

8.2. Configured the necessary settings to embed dashboards into external platforms , ensuring proper integration and interactivity.

9. **Comparison and Analysis:** Compared the Apache Superset and Metabase. Evaluated the two tools based on user interface, visualization options, ease of use, and performance.

10. **Project Conclusion:** Summarized findings on the comparative strengths and limitations of Apache Superset and Metabase for BI purposes.

11. **Future Exploration:** Outlined potential areas for further exploration and advanced capabilities of both Superset and Metabase to improve data insights and interactivity.

Work Products and Deliverables

- **Datasets**: A cleaned dataset on agricultural production in India (2013–2024), segmented by crop type and season.

- **Software Programs**: Docker configurations to set up and run Apache Superset and Metabase.

- **Documentation**: Detailed guides for installing Superset and Metabase, importing data, connecting to MySQL, and creating custom dashboards.
- **GitHub Repository**: Contains the dataset, Docker scripts, installation instructions, and code for building and embedding dashboards in both BI tools.

3 Dataset Overview

The dataset records agricultural production in India from 2013 to 2024, detailing crops such as food grains, cereals, pulses, oilseeds, and commercial crops, segmented by three main seasons: Kharif, Rabi, and Summer. It provides production values in lakh tonnes, or in bales for cotton and jute/mesta. The dataset, sourced from the *Department of Agriculture and Farmers Welfare*, contains three tables in three different sheets with a combined total of 34 Rows and 13 Columns with all text data types in default. The source file is dated on 25.09.2024.

Here's a sample view of the dataset:

Ministry of Agriculture & Farmers Welfare
Department of Agriculture & Farmers Welfare (DA&FW)
Final Estimate of Production of Food Grains

Dated : 25.09.2024
Source: DA&FW
Production in Lakh Tonnes

Crop	Season	Production									
		2013-14	2014-15	2015-16	2016-17	2017-18	2018-19	2019-20	2020-21	2021-22	2022-23	2023-24
Rice	Kharif	914.97	913.92	914.13	963.03	971.35	1020.40	1022.77	1052.08	1110.01	1105.12	1132.59
	Rabi	151.49	140.91	129.95	133.96	156.22	144.38	165.94	191.60	184.71	150.04	146.01
	Summer	@	@	@	@	@	@	@	@	@	102.40	99.65
	Total	1066.46	1054.82	1044.08	1096.98	1127.58	1164.78	1188.70	1243.68	1294.71	1357.55	1378.25
Wheat	Rabi	958.50	865.27	922.88	985.10	998.70	1035.96	1078.61	1095.86	1077.42	1105.54	1132.92
Maize	Kharif	171.45	170.14	160.53	189.19	201.18	194.14	194.29	215.55	226.81	236.74	222.45
	Rabi	71.14	71.59	65.14	69.81	86.34	83.02	93.37	100.92	110.49	116.90	120.28
	Summer	@	@	@	@	@	@	@	@	@	27.21	33.92
	Total	242.60	241.73	225.67	259.00	287.53	277.15	287.66	316.47	337.30	380.85	376.65
Barley	Rabi	18.31	16.13	14.38	17.47	17.81	16.33	17.22	16.56	13.71	19.13	16.99
Jowar	Kharif	23.93	23.00	18.16	19.64	22.74	17.35	16.97	19.86	15.98	14.80	15.09
	Rabi	31.49	31.45	24.22	26.04	25.30	17.40	30.75	28.26	25.52	23.22	31.89
	Summer	@	@	@	@	@	@	@	@	@	0.13	0.40
	Total	55.42	54.45	42.38	45.68	48.03	34.75	47.72	48.12	41.51	38.14	47.37
Bajra	Kharif	77.09	91.84	80.67	97.30	92.09	86.64	103.63	108.63	97.81	103.49	96.63
	Summer	$	$	$	$	$	$	$	$	$	10.82	10.53
	Total	77.09	91.84	80.67	97.30	92.09	86.64	103.63	108.63	97.81	114.31	107.16
Ragi	Kharif	19.83	20.61	18.22	13.85	19.85	12.39	17.55	19.98	17.01	16.91	16.70
Small Millets	Kharif	4.30	3.86	3.91	4.42	4.39	3.33	3.71	3.47	3.67	3.84	4.49
Shree Anna /Nutri Cereals	Kharif	125.15	139.31	120.96	135.21	139.07	119.71	141.85	151.95	134.47	139.04	132.90
	Rabi	31.49	31.45	24.22	26.04	25.30	17.40	30.75	28.26	25.52	23.22	31.89
	Summer	@	@	@	@	@	@	@	@	@	10.95	10.93
	Total	156.64	170.76	145.17	161.25	164.36	137.11	172.61	180.21	160.00	173.21	175.72
Nutri/Coarse Cereals	Kharif	296.60	309.45	281.49	324.40	340.25	313.84	336.15	367.50	361.28	375.78	355.35
	Rabi	120.94	119.17	103.74	113.32	129.45	116.75	141.34	145.74	149.73	159.25	169.16
	Summer	@	@	@	@	@	@	@	@	@	38.17	44.85
	Total	417.54	428.62	385.22	437.72	469.70	430.59	477.48	513.24	511.01	573.19	569.36

Figure 1: Dataset Overview

In the final dataset, multiple cleaning steps were performed to enhance data clarity, accuracy, and consistency.

First, missing data (represented by symbols like `"@"`, `"$"`, and `"#"`) was addressed by either imputing values or excluding irrelevant entries. Records for crops labeled as `'Jute ##'`, `'Mesta ##'`, `'Total Pulses'`, `'Cereals'`, `'Total Food Grains'`, and `'Total Oil Seeds'` were removed to eliminate redundancy.

For conversion consistency, the production values for crops like `'Jute & Mesta ##'` and `'Cotton#'` were updated from lakh bales to tonnes by applying a multiplier and converting to tonnes. The production column values were rounded to two decimal places for uniformity.

Crop names with symbols were clarified by updating `'Jute & Mesta ##'` to `'Jute and Mesta'`. Finally, wherever production values were missing for certain crops in specific seasons, a value of 0 was assigned to maintain data integrity across years.

Here's a sample view of the cleaned dataset:

A	B	C	D
crop	season	year_range	production
Rice	Kharif	2013-2014	914.97
Rice	Kharif	2014-2015	913.92
Rice	Kharif	2015-2016	914.13
Rice	Kharif	2016-2017	963.03
Rice	Kharif	2017-2018	971.35
Rice	Kharif	2018-2019	1020.4
Rice	Kharif	2019-2020	1022.77
Rice	Kharif	2020-2021	1052.08
Rice	Kharif	2021-2022	1110.01
Rice	Kharif	2022-2023	1105.12
Rice	Kharif	2023-2024	1132.59
Rice	Rabi	2013-2014	151.49
Rice	Rabi	2014-2015	140.91
Rice	Rabi	2015-2016	129.95
Rice	Rabi	2016-2017	133.96
Rice	Rabi	2017-2018	156.22
Rice	Rabi	2018-2019	144.38
Rice	Rabi	2019-2020	165.94
Rice	Rabi	2020-2021	191.6
Rice	Rabi	2021-2022	184.71
Rice	Rabi	2022-2023	150.04
Rice	Rabi	2023-2024	146.01
Rice	Summer	2013-2014	0

crop +

Figure 2: Cleaned Dataset Overview

4 Installation Steps of Apache Superset & Metabase

4.1 Installing Superset using Docker Compose

Step 1: Install Docker Desktop from here.

Step 2: Install Git from here.

Step 3: Search for apache superset in Docker, then -> Pull -> Image will be created -> Run it.

Step 4: Check that the Superset container is running in Docker containers.

Step 5: Go to the command line and type `docker --version` to verify if Docker is functioning on your local computer.

Step 6: Now, type the following four commands in sequence to set up Apache Superset, making sure you have copied your latest tag from Docker images. You can also find your latest tag (it will appear on top) and copy it from the Docker Hub Website.

Command lines to run in Command Prompt:

```
docker run -d -p 8080:8088 -e "SUPERSET_SECRET
   _KEY=mysuperset" --name superset apache/
   superset:<put your latest tag here>
```

```
docker exec -it superset superset fab create-
   admin --username admin --firstname Superset
    --lastname Admin --email admin@superset.
   com --password admin
```

```
docker exec -it superset superset db upgrade
```

```
docker exec -it superset superset load_
   examples
```

```
docker exec -it superset superset init
```

Step 7: Open your web browser and go to `http://localhost:8080/superset/welcome/`.

Step 8: When prompted, enter the credentials: Username: `admin`, Password: `admin`.

Superset Installation Done.

Note : Make sure you have sample charts and dashboards visible in your superset UI. If not then look out for possible errors in your docker desktop while setting up superset, check the tag or restart the system.

4.2 Installing Metabase using Docker Compose

Step 1: Install Docker Desktop from here.

Step 2: Install Git from here.

Step 3: Follow the command prompt lines to run in terminal `docker pull metabase/metabase:latest` after which it is visible in your Docker images. `docker run -d -p 3000:3000 --name metabase metabase/metabase` to run the Metabase container.

Step 4: This will launch the Metabase server on port 3000 by default, you can access the port at `http://localhost:3000`.

Metabase Installation Done.

The official documentaion of Metabase for further information is given here.

5 Step-by-Step Guide to Data Import

5.1 Connecting MySQL Database to Apache Superset

Step 1: Install MySQL from here.

Step 2: `Open cmd -> docker exec -it <superset-container-name> bash`

Step 3://Or find container name using docker ps

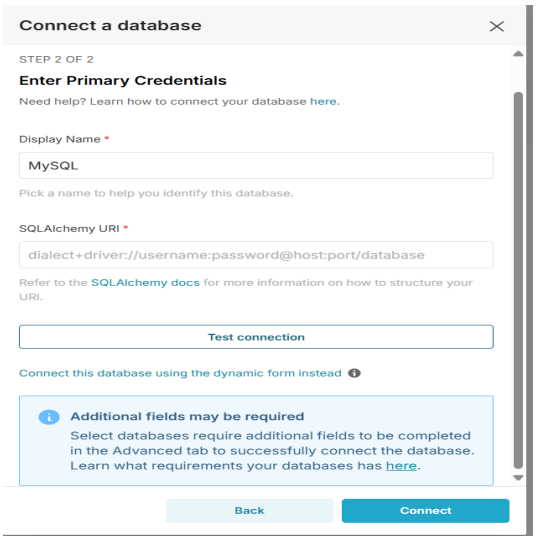
Step 4: Pip install mysqlclient (inside the container)

Step 5: Exit

Step 6: Go to the top-right corner of the Superset window, click on the `Settings` drop-down menu, and select **Database Connections**.

Step 7: Choose the **MySQL** option, then connect using the SQLAlchemy URI. Enter the following URI format:

```
mysql://<database_user>:<database_password>@host.
   docker.internal:3306/<your_schema>
```



Step 8: Test and Connect Here, replace only `<database_user>`, `<database_password>`, and `<your_schema>` with your specific details. Leave the rest of the URI as it is.

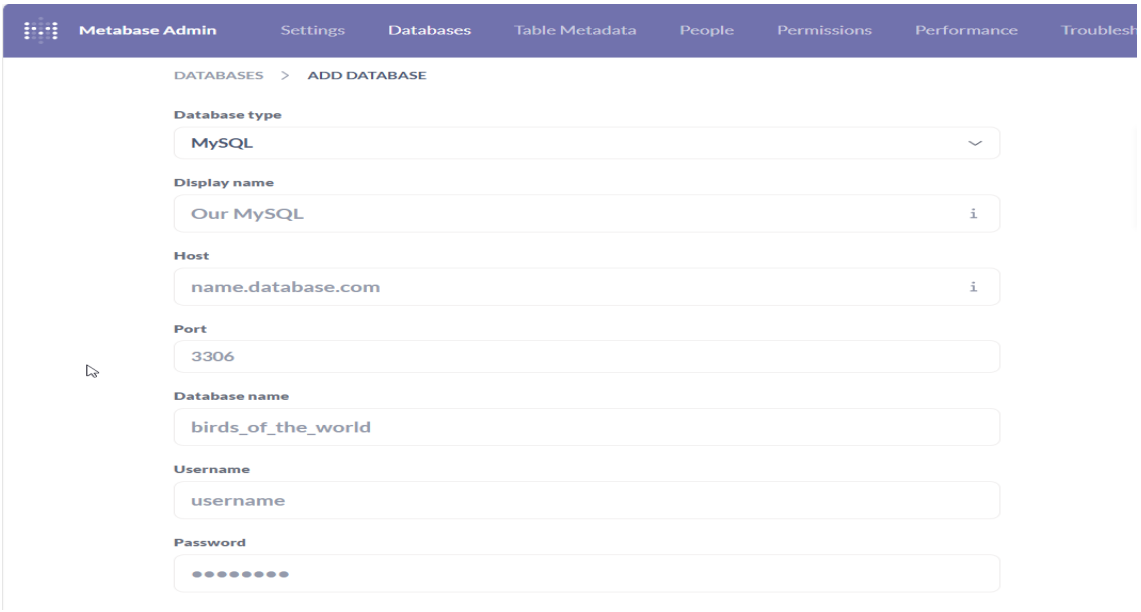
DATABASE CONNECTION DONE.

Note: If test connection does not work check for necessary driver installation in the documentation.

5.2 Connecting MySQL Database to Metabase

Step 1: Under the settings options, scroll to **Admin Settings** and click on **Add a Database**.

Step 2: Enter all the necessary credentials to connect the desired schema to the Metabase database.



Step 3: Or use the **Advanced** option to connect MySQL databases using a JDBC connection string if needed.

Step 4: Save it.

DATABASE CONNECTION DONE.

Note: MySQL is running locally and not inside a Docker container, so we need to use `host.docker.internal` as the hostname when connecting to Metabase (which is running inside a Docker container). This refers to the host machine, as MySQL is not present inside the Docker container.

6 Dashboard

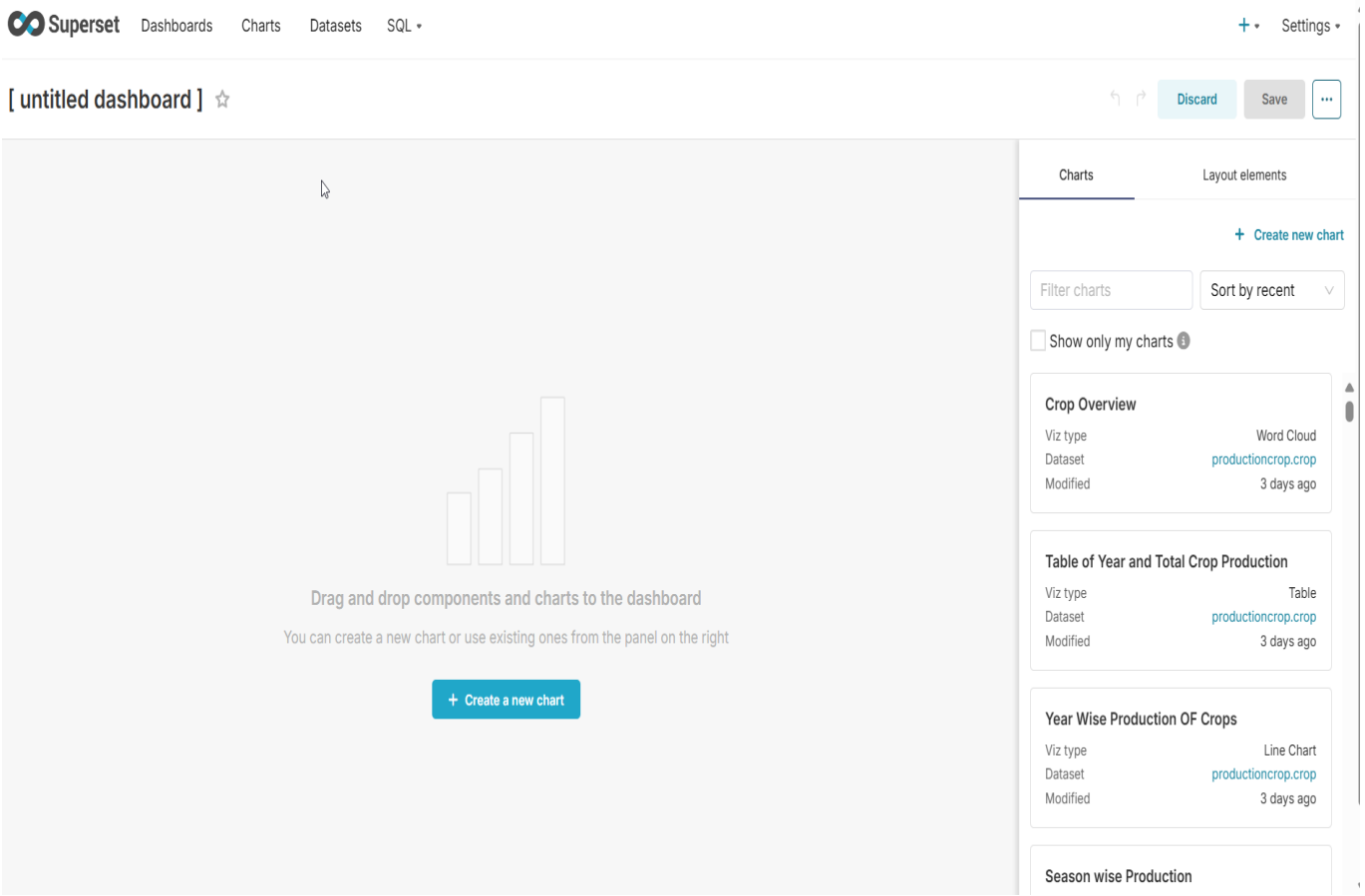
A dashboard is a visual representation of multiple sets of data. In the world of Apache Superset and Metabase, a dashboard is a collection of charts. Dashboards have the unique capability to tell a story by combining different types of charts to form a narrative. A dashboard can empower teams to present powerful data-based presentations as well as enable organizations to monitor information based on dynamic data.

6.1 Dashboard Creation in Apache Superset:

Step 1: Go to the top-left corner of the Superset window, click on `Dashboards`.

Step 2: To create new dashboard, go to `'+ Dashboard'`.

Step 3: The process of creating a dashboard in Superset is briefly explained below.



The above picture explains the layout and functionality of the "Dashboard Creation" interface in Superset, which enables users to assemble various charts and visualizations into a cohesive dashboard.

- **Top Navigation**: Access *Dashboards*, *Charts*, *Datasets*, and *SQL*.
- **Dashboard Title**: "[untitled dashboard]" (click to rename).

- **Save/Discard**: Save or discard dashboard changes.
- **Main Workspace**: Drag and drop charts here; option to create new charts.
- **Right Sidebar**: Lists available charts with details (e.g., name, type).
- **Drag and Drop**: Add, arrange, and resize charts for a custom layout.

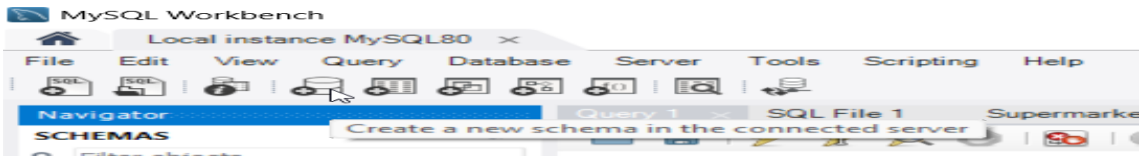
6.2 Example : Building an Agricultural Production Dashboard in Apache Superset

This section provides an example of creating a dashboard in Apache Superset to analyze agricultural production in India. The dashboard will showcase various visualizations and metrics related to crop production trends, season-wise yield to offer insights into India's agricultural sector.

Step-by-Step Guide to Creating an Agricultural Production Dashboard for India:

Step 1: Open MySQL Workbench.

Step 2: Create a new schema.



Create a new schema `->` Name the schema and then click `Apply` button `->` Click `Apply` button `->` Click `Finish` button

New schema, named as `productioncrop` is created in MySQL.

Step 3: Import table in MySQL.

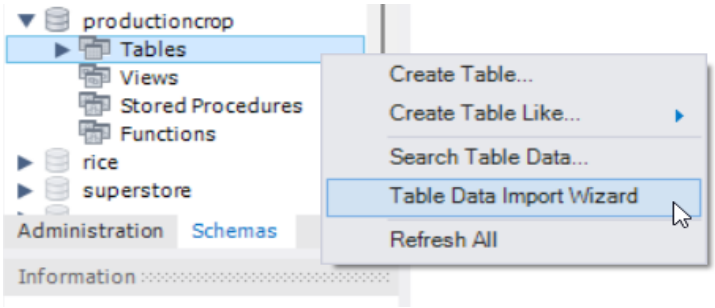
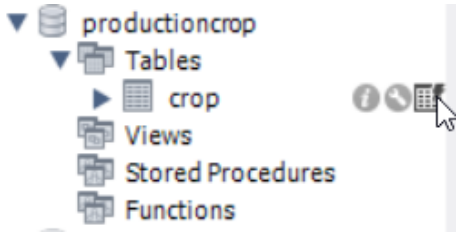


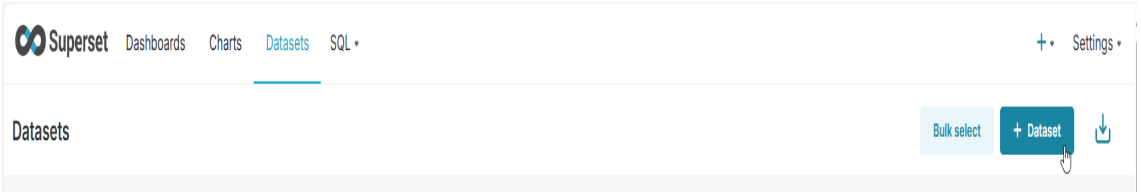
Figure 3: Click `Table Data Import Wizard` button

Click `Table Data Import Wizard` button-> Select the file to import and Click `Next` button->Click `Next` button-> Check the column and field type and click `Next` button->To import data click `Next` button-> click `Next` button -> click `Finish` button

Table is now successfully imported under the `productioncrop` schema in MySQL .



Step 4: Connecting database to Superset.



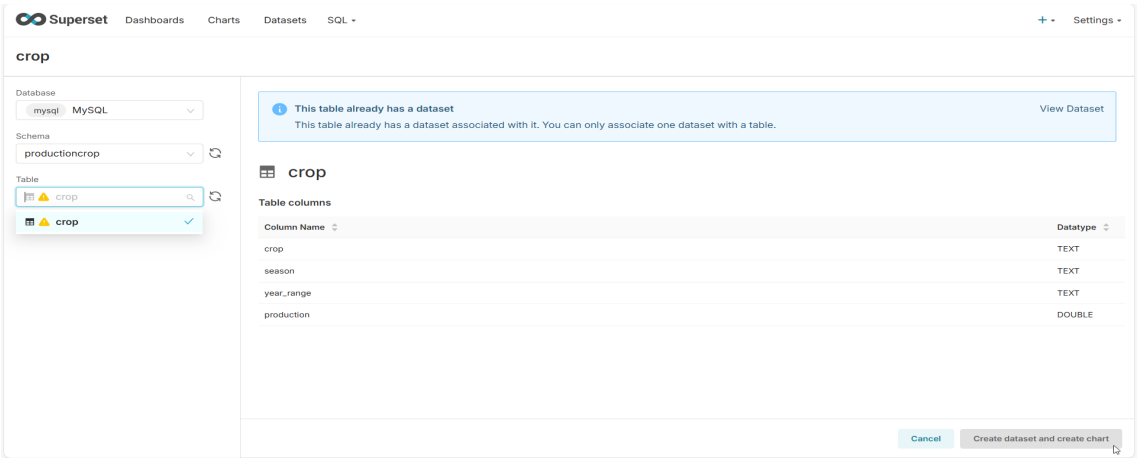
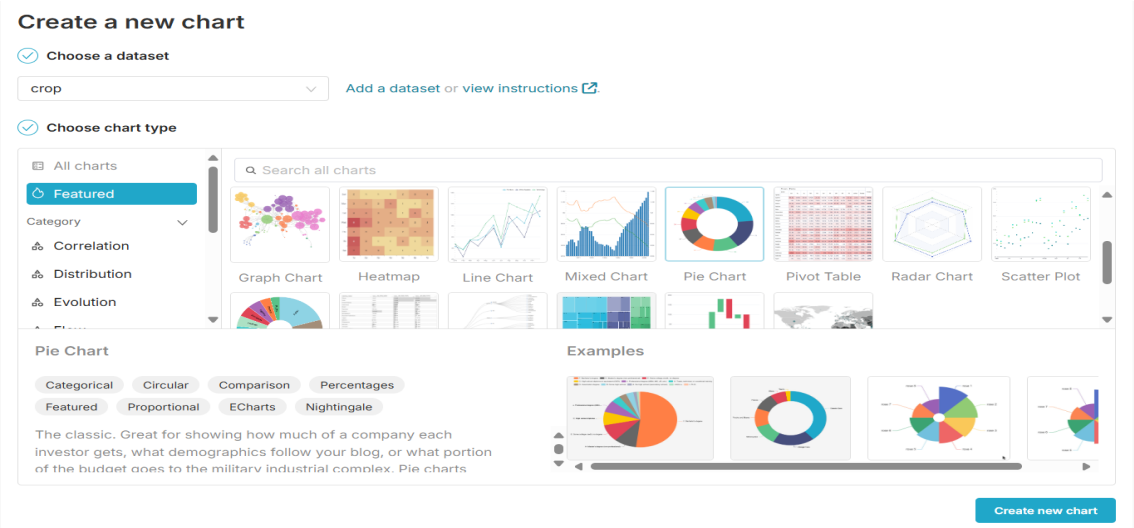


Figure 4: Select your `Database`, then select `schema`, then select `Table`, then click on `Create dataset and create chart`

.

Dataset is created successfully.

Step 5: Create charts on Superset.



Choose a `Dataset`, then choose a `chart type`, then click on `Create new chart`

Now we create pie chart which represents various crop production values in India.Details is explained below.

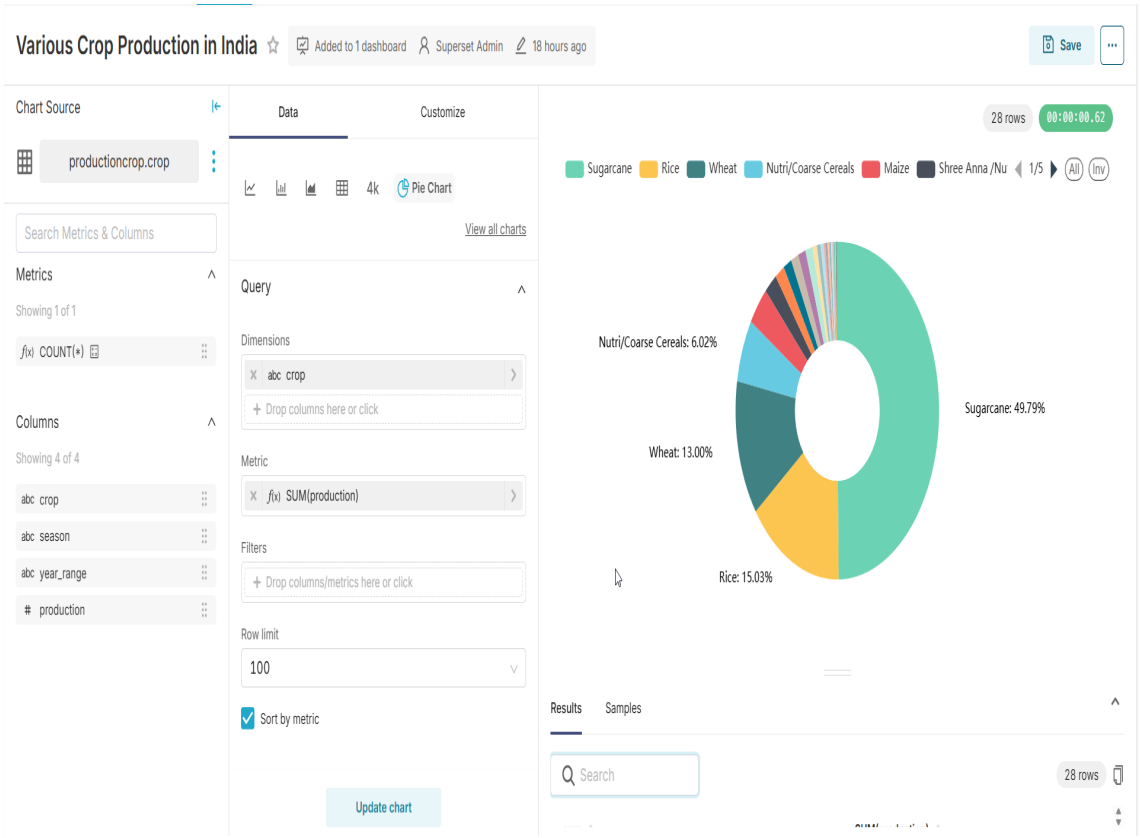


Chart Title

The title **"Various Crop Production"** is displayed at the top, indicating the purpose of this chart.

Chart Source

On the left, the **Chart Source** section lists the dataset used, which in this case is `productioncrop.crop`.

Metrics and Columns

- **Metrics:** Shows the available metric, which is `COUNT(*)`.
- **Columns:** Lists the columns in the dataset (crop, season, year_range, and

production), which can be used for visualization.

Query Panel

- **Dimensions:** The dimension chosen here is `crop`, which means each segment of the pie chart represents a different crop type.
- **Metric:** The metric is `SUM(production)`, aggregating the total production value for each crop.
- **Row Limit:** Limits the number of rows shown in the chart to 100, sorted by the metric value.

Pie Chart Visualization

The right side displays the pie chart visualization, with each slice representing a crop and its proportion of total production. The chart also shows the percentage breakdown of each crop's contribution. In this chart, sugarcane has the largest share, comprising 49.79% of total production.

Save and Update Options

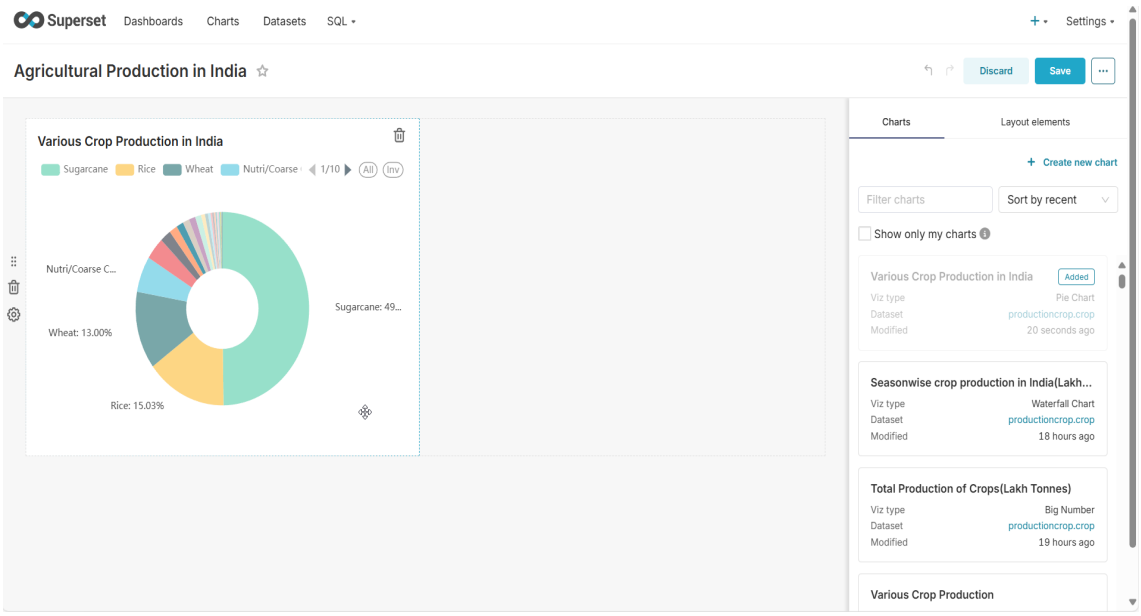
The **"Save"** button on the top-right allows users to save the chart. The **"Update Chart"** button at the bottom of the query panel applies any changes made to the chart configuration.

Conclusion

In summary, a pie chart of crop production, highlighting the relative production values of different crops based on the selected dataset and metrics.

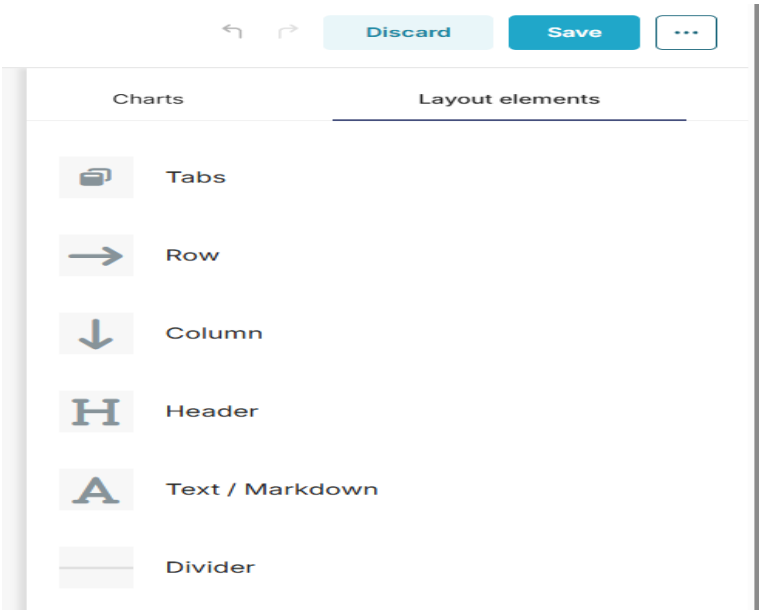
We make various types of charts, which are given in the Appendix section.

Step 6: Dashboard Creation on Apache Superset

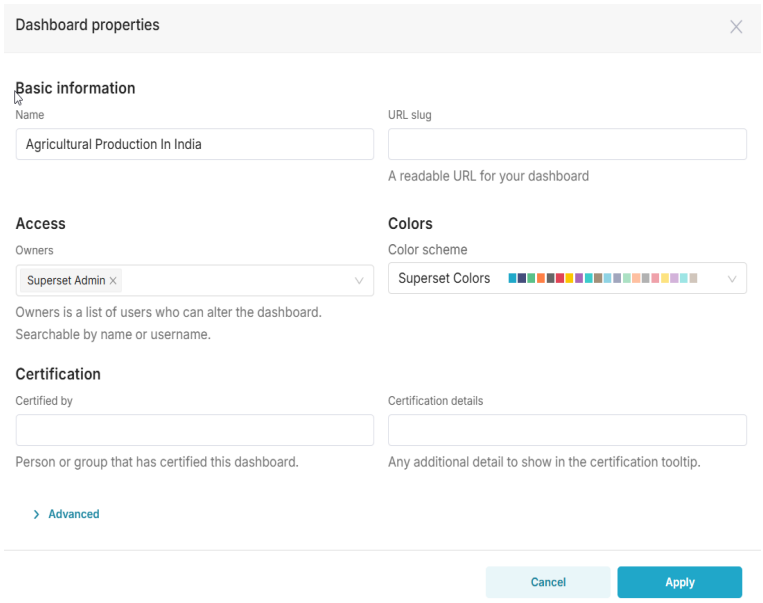


Selects the required charts in the dashboard.

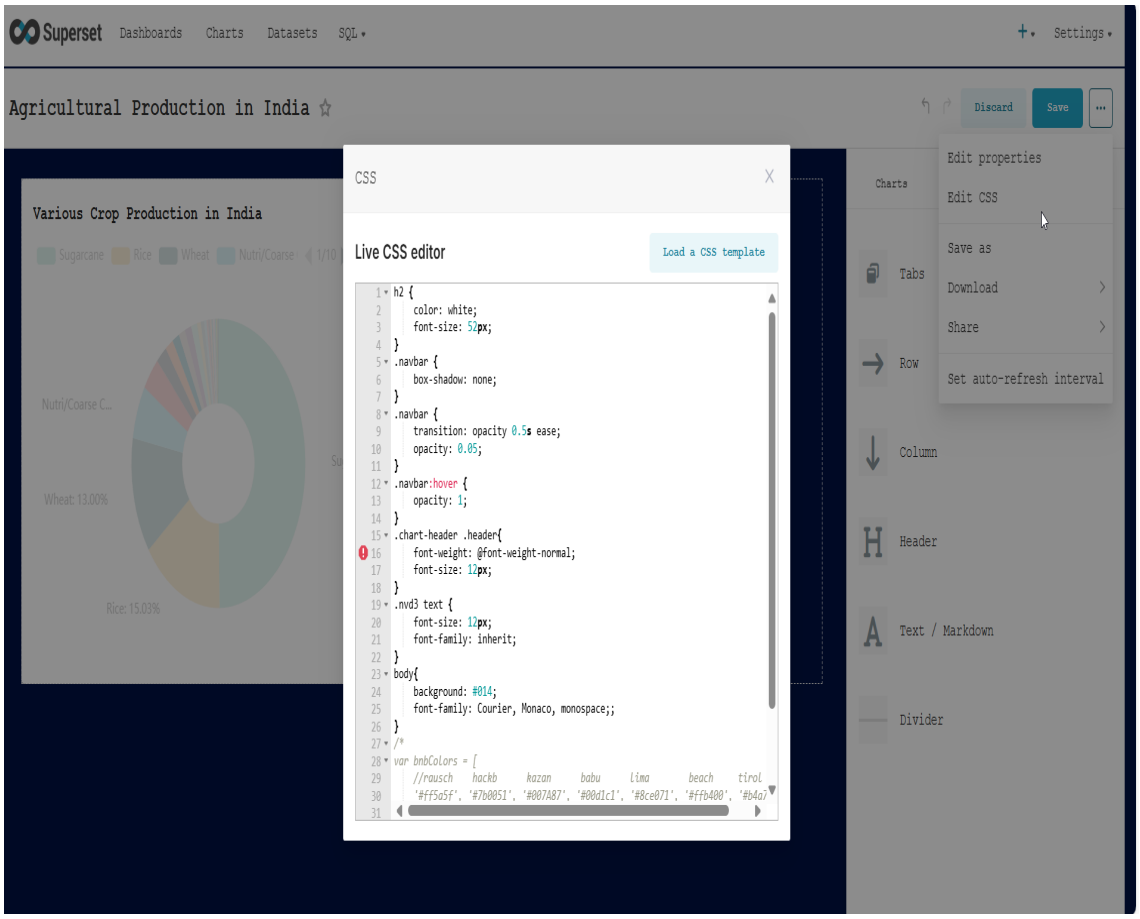
Step 7: We can explore layout elements. Like adding tabs, header etc.



Step 8: We can also customized the dashboard properties as well.



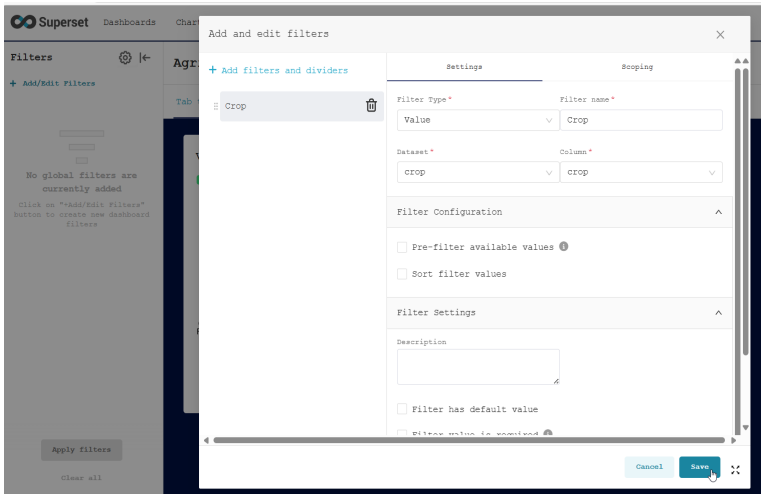
Step 9: Edit CSS.



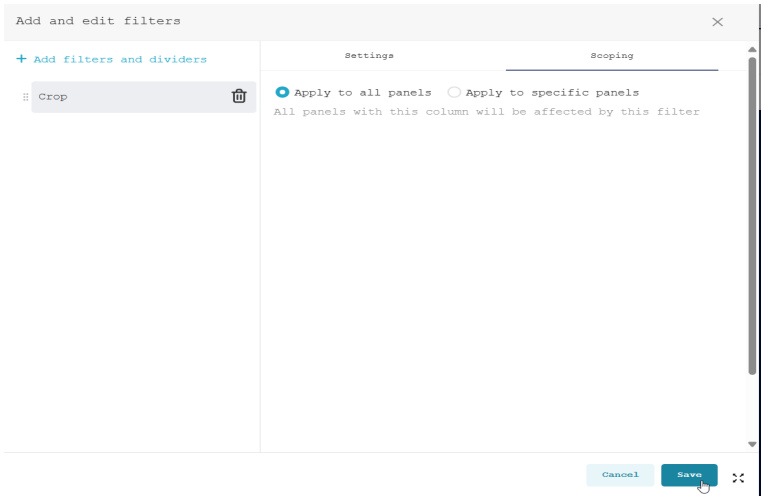
Customize your dashboard as you want.

Step 10: Save the dashboard.

Step 11: We can select filters if we want.



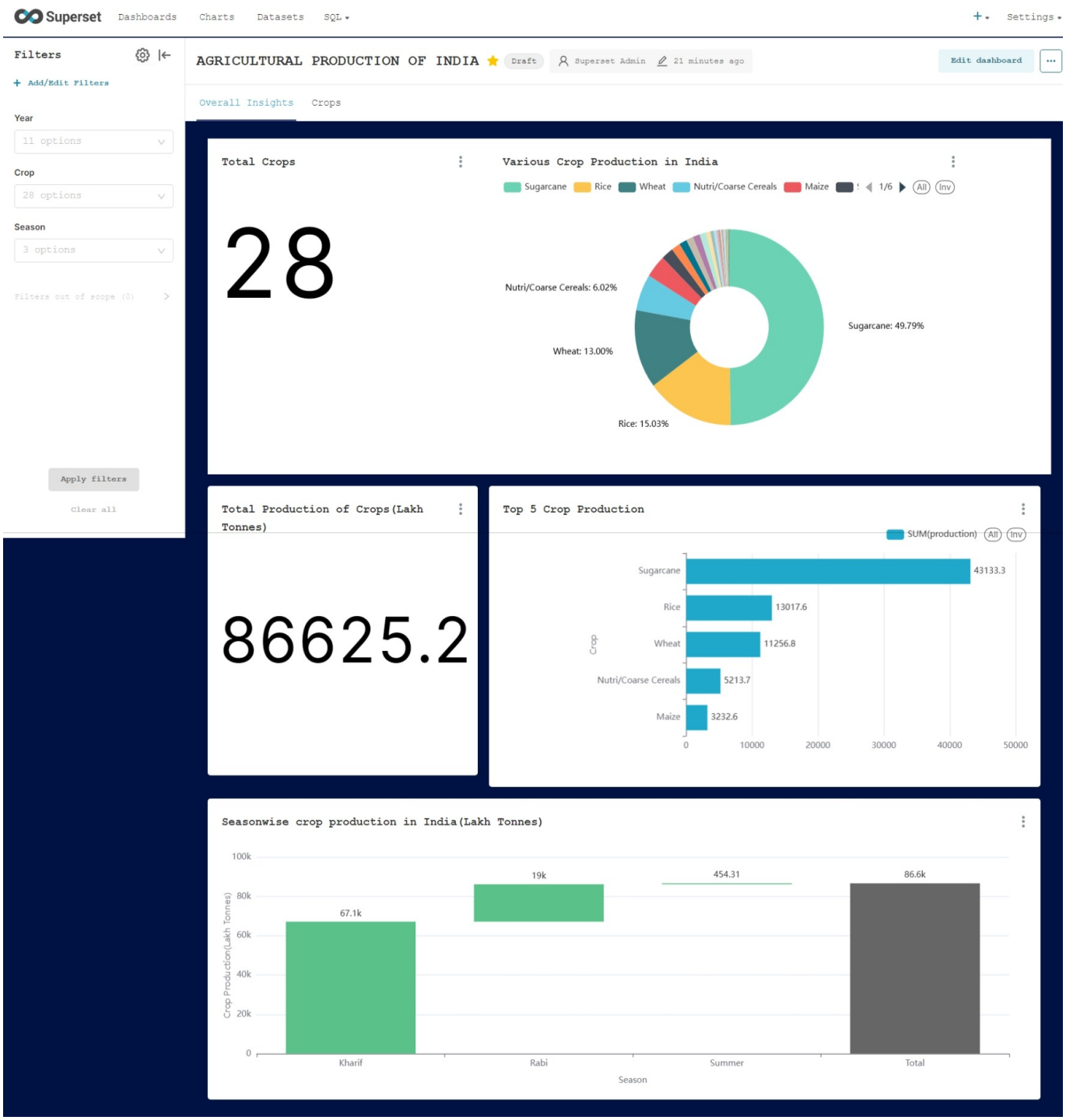
Go to filter settings -> Filter name -> Add Dataset -> Select Column.Customize your filters as you want.



Go to scooping -> Select `Apply to all panels` or `Apply to the specific panels` -> Save

Filters is now created.

AGRICULTURAL PRODUCTION IN INDIA DASHBOARD(USING APACHE SUPERSET)



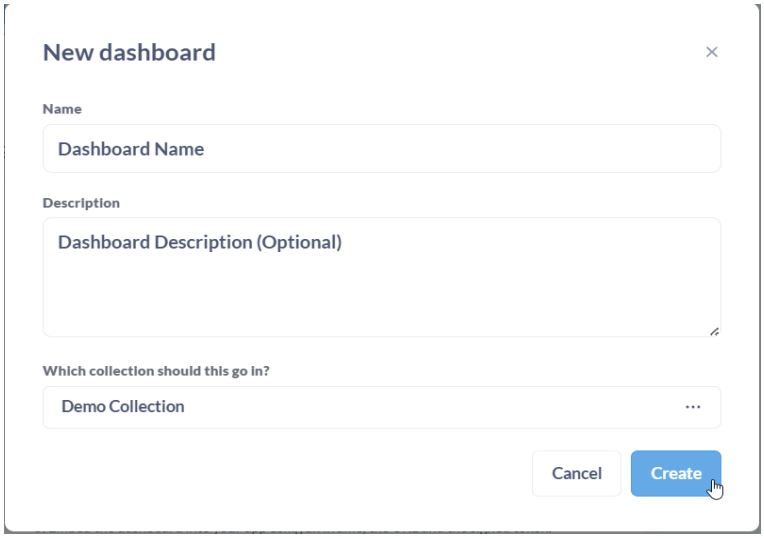
This is the overall insights tab of Agricultural Production in India.User can use the filters as wants.

This is the crop tab which shows the overall view of crops in India.User can use the filters as wants.

6.3 Dashboard Creation in Metabase:

Step 1: Go to the top right of the screen, click the `+ New -> Dashboard`.

Step 2: Give `New dashboard name -> description(Optional)->` Choose the `collection` where the dashboard should go `-> Click create`.



Now the dashboard is created.Now user can create charts and customize there dashboard as they want.

6.4 Example : Building an Agricultural Production Dashboard in Metabase

This section provides an example of creating a dashboard in Metabase to analyze agricultural production in India. The dashboard will showcase various visualizations and metrics related to crop production trends, season-wise yield to offer insights into India's agricultural sector.

Step-by-Step Guide to Creating an Agricultural Production Dashboard for India

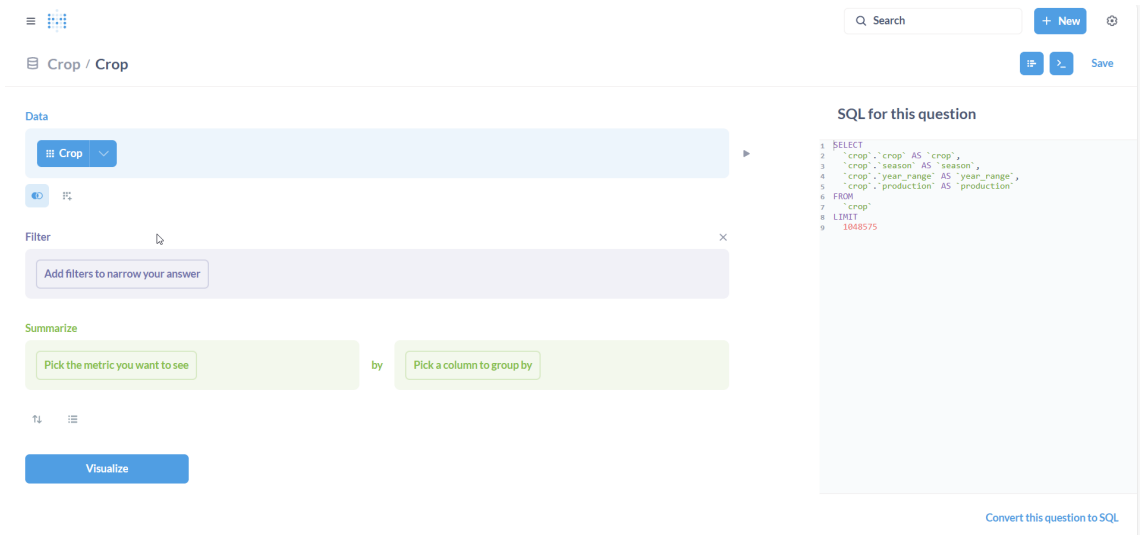
Step 1: Open `MySQL -> Create New Schema -> Click Table -> Table Data Import Wizard -> Browse` file to import `Next -> Finish`.

Detailed explanation is given in the **Step1, Step2 and Step 3**.

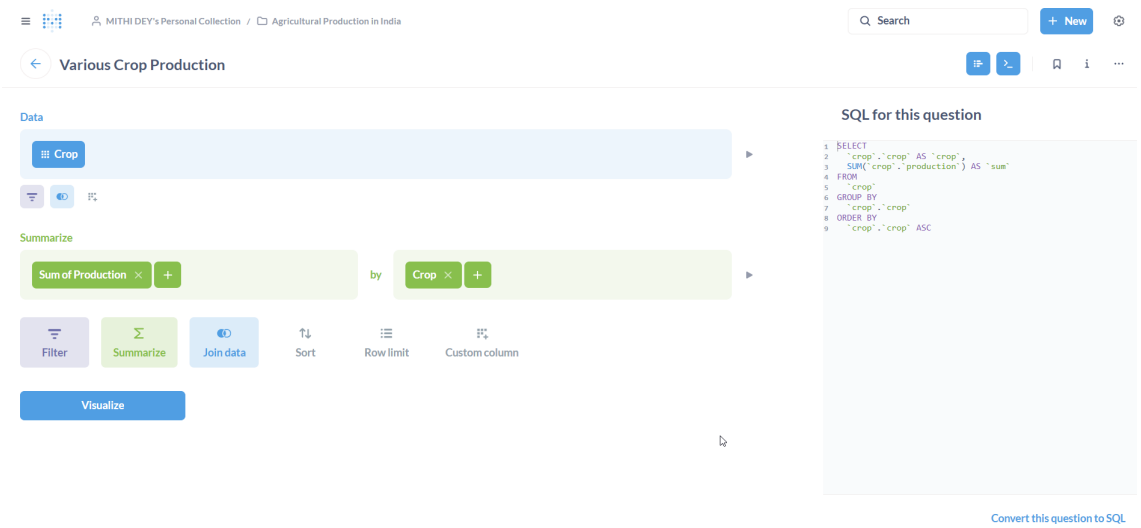
Step 2: Connecting database to Metabase. Follow this step.

Step 3: Create charts on Metabase.

Go to `+New -> Question ->` Select your `dataset -> Filter(Optional) -> Summarize -> Group by -> Visualize`



Now we create pie chart which represents various crop production values in India. Details is explained below.



Explanation of the Crop Production Data Query Page

Data Source

The selected data source is a table named `Crop`, which likely contains information on various crops and their production statistics.

Summarize Section

This section is set to compute the `Sum of Production` for each unique `Crop`. This means the query will calculate the total production for each type of crop in the dataset.

SQL Query Output

On the right side, the tool displays the SQL code generated based on the setup:

- **SELECT**: It selects the `crop` field and computes the `SUM` of `production`.
- **GROUP BY**: It groups the results by each crop to calculate total production for each.

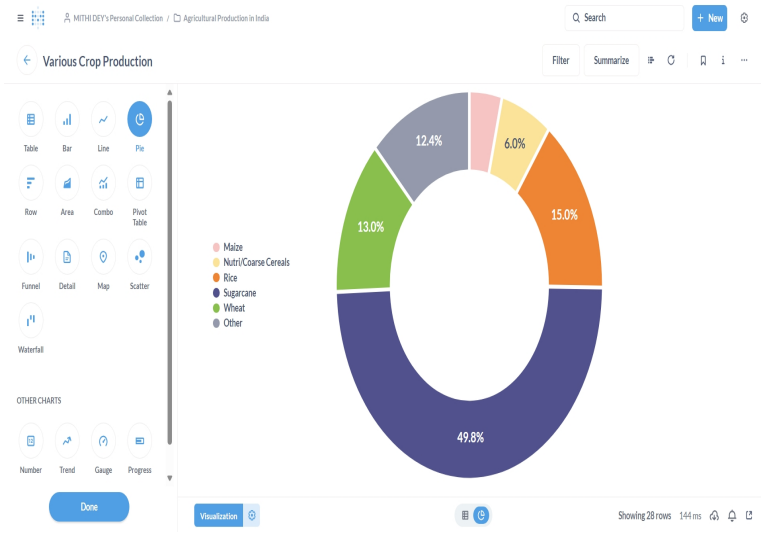
- **ORDER BY**: The results are ordered alphabetically by crop name.

Options for Further Customization

- **Filter**: Allows filtering data based on specific criteria.
- **Join Data**: Enables joining with other tables if additional data is required.
- **Sort**: For ordering results differently.
- **Row Limit and Custom Column**: To limit the number of rows displayed or to add a custom calculated column.

The `Visualize` button allows us to see the results in a chart or table format based on these configurations.

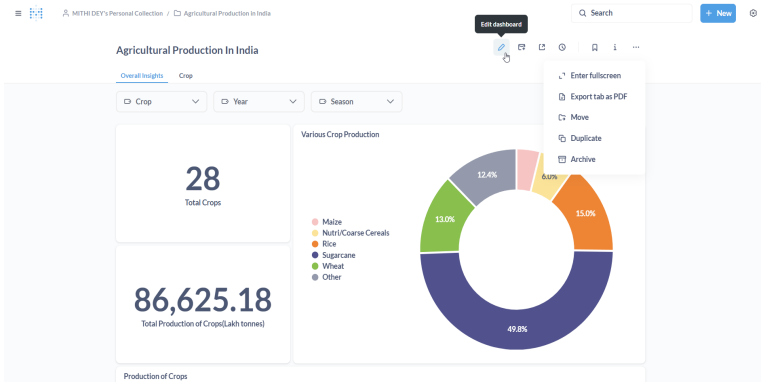
We will create a pie chart.



We can also customize our chart as we want. To customize our chart go to `Display` option.

Chart is created now. Now, `Save` the chart `->` Give `New chart name ->` `description(Optional)->` Choose the `collection` where the dashboard should go `->` `Click save ->` If you want to add the charts direct to the dashboard then click `Yes Please!`, when the pop up window came.

Step 4: Dashboard Creation on Metabase.



Selects the required charts from the collection or create a new question.

Step 5: Add filters, add sections, add headings. We can customize the dashboard as required.

AGRICULTURAL PRODUCTION IN INDIA DASHBOARD (USING METABASE)

This dashboard presents an overview of agricultural production in India. It shows 28 types of crops with a total production of 86,625.18 lakh tonnes. The pie chart

highlights the main contributors, including Sugarcane 49.8% . The bar charts further break down total production, emphasizing Sugarcane, Rice, and Wheat as the top crops.User can use the filters as wants.

The dashboard analyzes agricultural production in India, showing that Kharif season yields the highest output at 67,132.22 lakh tonnes. A line chart indicates a steady increase in overall production from 2013–2014 to 2023–2024, reaching 9,069.09 lakh tonnes. The bubble chart compares production volumes for key crops like Rice and Wheat, while a summary table reinforces the upward trend in total production across years.User can use the filters as wants.

7 Embedding Superset & Metabase dashboard into HTML

- **Step 1:** Create an HTML file. Example is given below:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
       scale=1.0">
    <title>Embedded Superset Dashboard</title>
    <style>
        /* Style for the iframe */
        .dashboard-iframe {
            width: 100%;
            height: 100vh;
            border: none;
        }
        /* Style for the button */
        .switch-button {
            margin: 10px;
```

```
            padding: 10px 20px;
            font-size: 16px;
            cursor: pointer;
        }
    </style>
</head>
<body>
    <h1>Agricultural Production in India</h1>
    <button class="switch-button" id="switchButton">Switch to
       Metabase</button>
    <!-- Embed dashboard using iframe -->
    <iframe id="dashboardIframe" class="dashboard-iframe"
        src="http://localhost:8080/superset/dashboard/p/WpOON0DZVKm
           /">
    </iframe>

    <script>
        let isSuperset = true;

        document.getElementById('switchButton').onclick = function
           () {
            const iframe = document.getElementById('dashboardIframe
               ');
            if (isSuperset) {
                iframe.src = "http://localhost:1234/public/
                   dashboard/681bc3c6-3856-4885-87f6-7d39487a6757";
                    // Metabase URL
                this.innerText = 'Switch to Superset';
            } else {
```

```
                iframe.src = "http://localhost:8080/superset/
                    dashboard/p/WpOONODZVKm/"; // Superset URL
                this.innerText = 'Switch to Metabase';
            }
            isSuperset = !isSuperset;
        };
    </script>
</body>
</html>
```

This HTML code embeds a Superset dashboard in an iframe and provides a button to switch between Superset and Metabase dashboards. The button toggles the iframe's src URL, switching the displayed dashboard between the two BI tools. Styling is included for the iframe and the button.

7.1 Steps to Embed Superset Dashboard into HTML

Step 1: Create the Superset Configuration File:

- Create a file named `superset_config.py` on your Desktop.

Step 2: Update Permissions in Superset:

- Create an HTML file.
- Add necessary permissions for the `Gamma` role in Superset, including:
  - Schema access
  - Datasource access
  - CSS permissions

Step 3: Mount the Configuration File in Docker:

- Use the following command to mount the file:

```
docker cp /Desktop/superset_config.py
<superset_container_name>:/app/pythonpath/superset_config.py
```

Step 4: Verify the File Has Been Copied:

- Access the Docker container:

```
docker exec -it <container_name_or_id> /bin/bash
```

Step 5: Navigate to the Desired Path Inside Docker:

- Use the `ls` command to list files.
- Exit the shell using `exit`.

Step 6: Restart the Superset Container:

- Use the command:

```
docker restart <superset_container_name>
```

Step 7: Update `superset_config.py` with Feature Flags:

- Add the following content to `superset_config.py`:

```
FEATURE_FLAGS = {
"EMBEDDED_SUPERSET": True,
}
ENABLE_PROXY_FIX = True
SESSION_COOKIE_SAMESITE = None
PUBLIC_ROLE_LIKE_GAMMA = True
AUTH_ROLE_PUBLIC = 'Gamma'
WTF_CSRF_ENABLED = False
TALISMAN_ENABLED = False
```

Superset dashboard embedding is done.

7.2 Steps to Embed Metabase Dashboard into HTML

Step 1: Enable embedding in Metabase.

- Go to the *Admin* settings of Metabase and turn on the embedding feature.

Step 2: Navigate to your desired dashboard.

- Click on the *Share* option and then select *Embed*.

Step 3: Generate and copy the embed link.

- Obtain the *Public embed* link from the options provided and add this link to your HTML file.

Step 4: Understand potential browser restrictions.

- If you open the HTML file directly, browser restrictions like CORS (Cross-Origin Resource Sharing) could prevent the dashboard from loading correctly. This would require changing configuration settings, which can be complex.

Step 5: Use a local web server as a workaround.

- By serving the HTML file through a local web server, both the Metabase dashboard and the HTML file will be seen as coming from the same host, avoiding these restrictions.

Step 6: Start a local web server using Python.

- Open the command prompt and run:

```
python -m http.server 8000
```

Step 7: Access the HTML file through the browser.

- In your web browser, go to:

```
http://localhost:8000/name_of_the_html_file
```

- Replace `name_of_the_html_file` with the actual name of your file.

This method ensures that the dashboard is properly loaded without CORS issues.

8 Apache Superset vs Metabase

8.1 Data Sources

Category	Metabase	Apache Superset
SQL databases & engines	16	43
NoSQL databases	1 (MongoDB)	0 (possible through Trino / Presto)

Table 1: Comparison of Data Sources in Metabase and Superset

8.2 No-code Query Builder

Feature	Metabase	Superset
Support common SQL operations (Select, Filter, GroupBy, etc)	✓	✓
Preview visualization during iteration	✓	✓
Preview generated SQL query during iteration	✓	✓
View data transformation results as a table	✓	✓
Use virtual datasets to build charts	✓	✓
Ability to join tables	✓	✓

Table 2: No-code Query Builder Comparison: Metabase vs Superset

8.3 Visualization Features

Feature	Metabase	Superset
Common charts (pie, line, bar, etc)	✓	✓
Funnel visualizations	✓	✓
Sankey diagrams	✗	✓
Network visualization	✗	✓
Clear documentation on adding your own visualizations	✗	✓
Number of default charts	17	62 (and growing)
Number of geospatial visualizations	3	10
Configuration options (e.g., tooltips, colors)	Limited	Robust
Custom charts (via plugins)	✗	✓

Table 3: Visualization Features Comparison: Metabase vs Superset

8.4 Advanced SQL Editor

Feature	Metabase	Superset
Export query results as CSV	✓	✓
Database metadata explorer	✓	✓
Linting and auto-complete	✓	✓
Support for variables in SQL queries	✓	✓
Multi-tab workflow	✓	✓
Save query for re-use	✓	✓
Semantic layer: save queries as virtual datasets	✓	✓
Semantic layer: define custom metrics	✓	✓
Semantic layer: calculated columns	✓	✓

Table 4: Advanced SQL Editor Features Comparison between Metabase and Superset

Both Metabase and Superset ship with a powerful SQL editor and a lightweight semantic layer.

- Aggregate values across multiple columns and publish as **Metrics**.
- Metrics can be certified as authoritative by a specific user.
- Transform specific columns and publish as **Calculated Columns**.
- Write arbitrary SQL queries and publish as a **Virtual Dataset**.

In the no-code chart builder (**Explore**), metrics, columns, and virtual datasets all inherit the power that physical database tables have in Superset.

Feature	Metabase	Superset
Basic datetime, value, and range filters	✓	✓
Dashboard templating	✓	✓
Custom theming	✓	✓
Dashboard can contain charts from multiple data sources	✓	✓
Cross-filtering	✓	✓
Data drilling (drill-down, drill by)	✓	✓
Organization and configuration	Limited	Robust

Table 5: Feature Comparison between Metabase and Superset

9 Conclusion and Recommendation

Apache Superset and Metabase are both powerful open-source Business Intelligence (BI) tools, but they cater to different users. Superset is ideal for technical teams needing advanced analytics, complex visualizations, and security, making it best suited for data engineers and analysts. Metabase, on the other hand, is user-friendly and suitable for quick insights, perfect for smaller businesses or non-technical teams.

Recommendation

Choose Superset if you need flexibility, scalability, and advanced visualizations; choose Metabase if you want an accessible, no-fuss BI tool for fast insights.

Best Use Cases

- **Apache Superset:**
  - **Enterprise Analytics:** Ideal for large organizations needing in-depth analysis across multiple departments.
  - **Geospatial Insights:** Great for companies needing location-based data, such

as logistics or retail.

- **Metabase:**
  - **Small Business Reporting:** Perfect for small e-commerce businesses or startups tracking sales and marketing metrics.
  - **Departmental Analytics:** Non-technical teams in larger companies can use it for independent, easy data exploration.

In summary, use Superset for scalability and Metabase for simplicity.

10 Future Exploration of Apache Superset and Metabase

Apache Superset

- **Advanced Visualizations**: Ideal for complex datasets with options like time series, geospatial, and heatmaps.
- **SQL Lab**: Custom SQL queries enable tailored analyses and data filtering.
- **Real-Time Data**: Supports real-time updates for timely insights on metrics like weather impacts.
- **Dynamic Dashboards**: Auto-updates based on the latest data for continuous monitoring.
- **Role-Based Access**: Secure data handling with restricted access to sensitive information.

Metabase

- **No-Code Query Builder**: Enables data exploration without SQL, making it accessible for non-technical users.
- **Embedded Analytics**: Integrate dashboards into apps, providing direct access to data insights.
- **Alerts and Notifications**: Set alerts for key metrics to stay informed on critical changes.
- **Collaboration**: Easy sharing for collaborative analysis and strategic planning.
- **Scheduled Reports**: Routine updates keep stakeholders informed for data-driven decisions.

Both tools support scalable, actionable insights essential for modern, data-driven strategies.

11 Appendix

- Click here for GitHub Link.
- Apache Superset Quick Start Guide: Develop interactive visualizations by creating user-friendly dashboards
- Apache Superset Documentation
- Metabase Documtation
- Guides and tutorials on Metabase
- Start Data Engineering : Apache Superset Tutorial
- Preset Documentation.
- My Dashboard-http://localhost:8000/superset.html

- **Charts Created in Apache Superset**

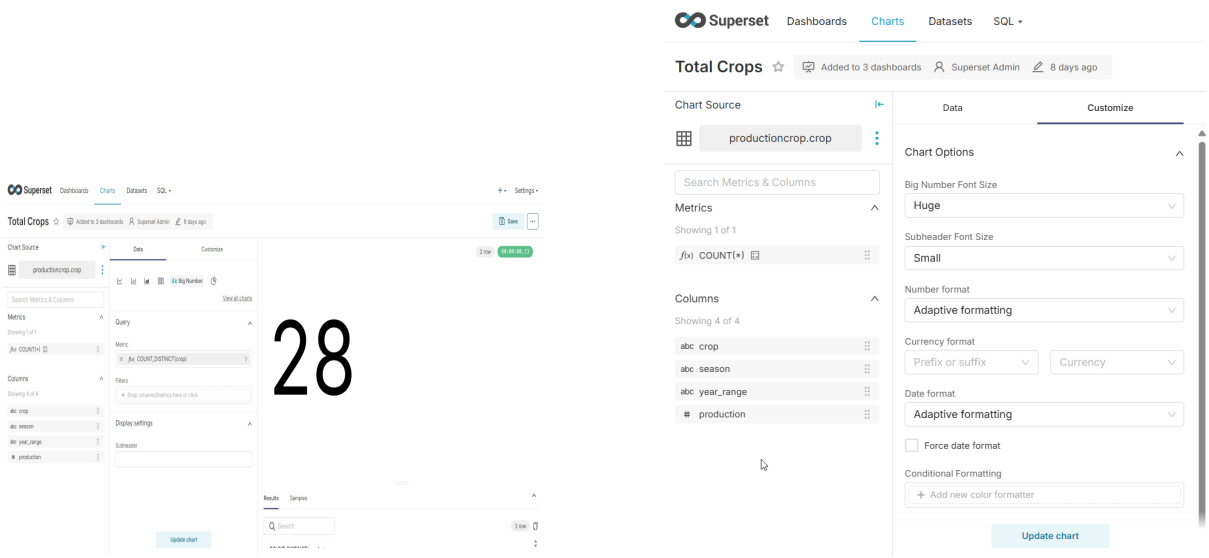


Figure 5: Total Crops

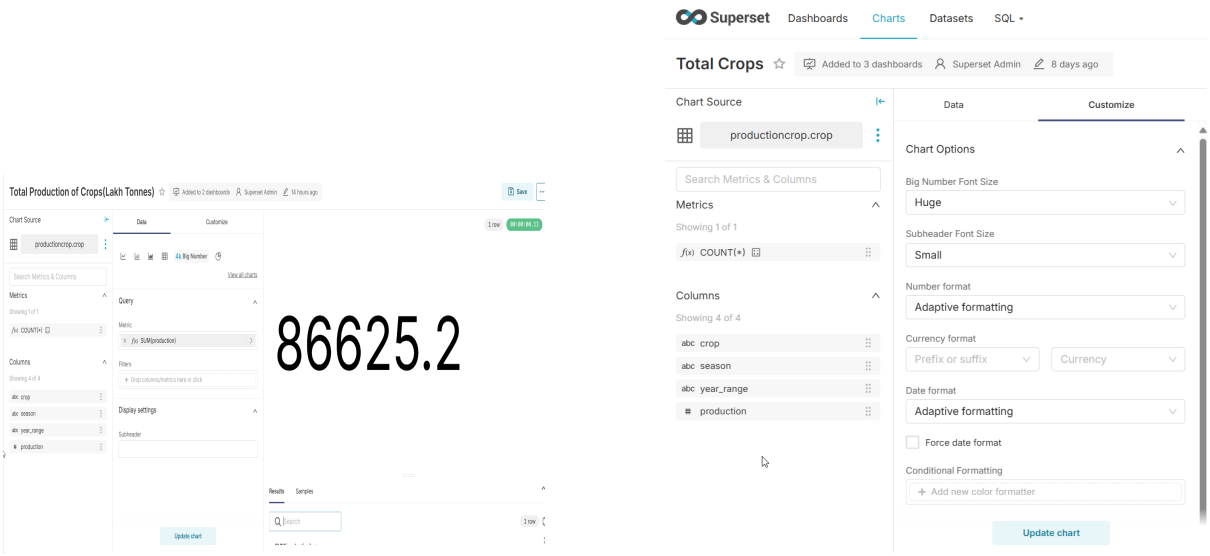


Figure 6: Total Production of Crops (Lakh Tonnes)

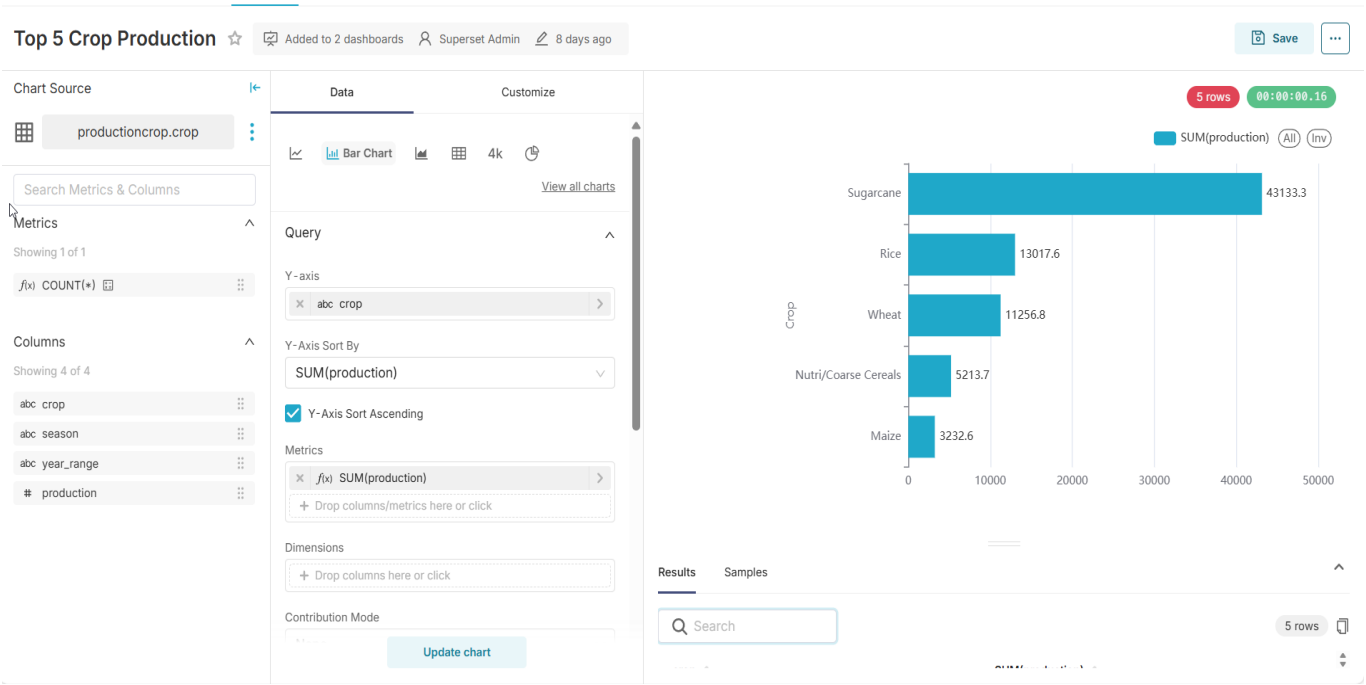


Figure 7: Top 5 Crops

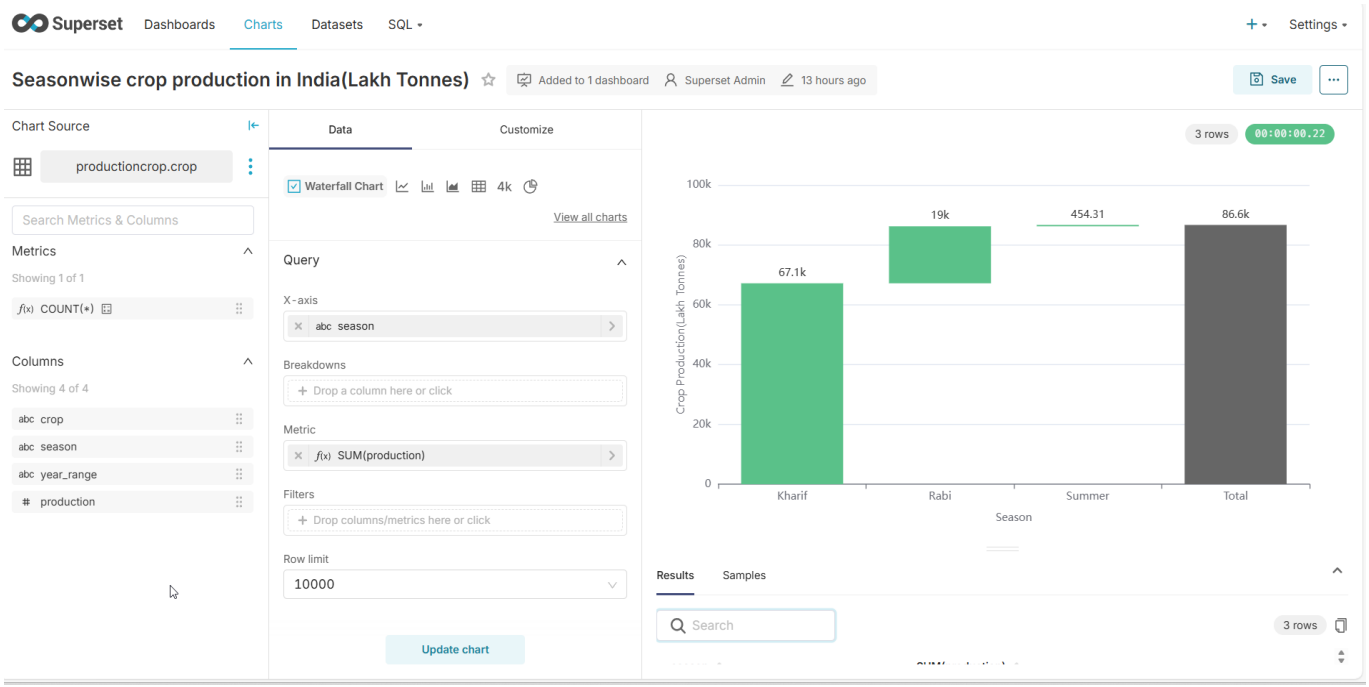


Figure 8: Seasonwise Production

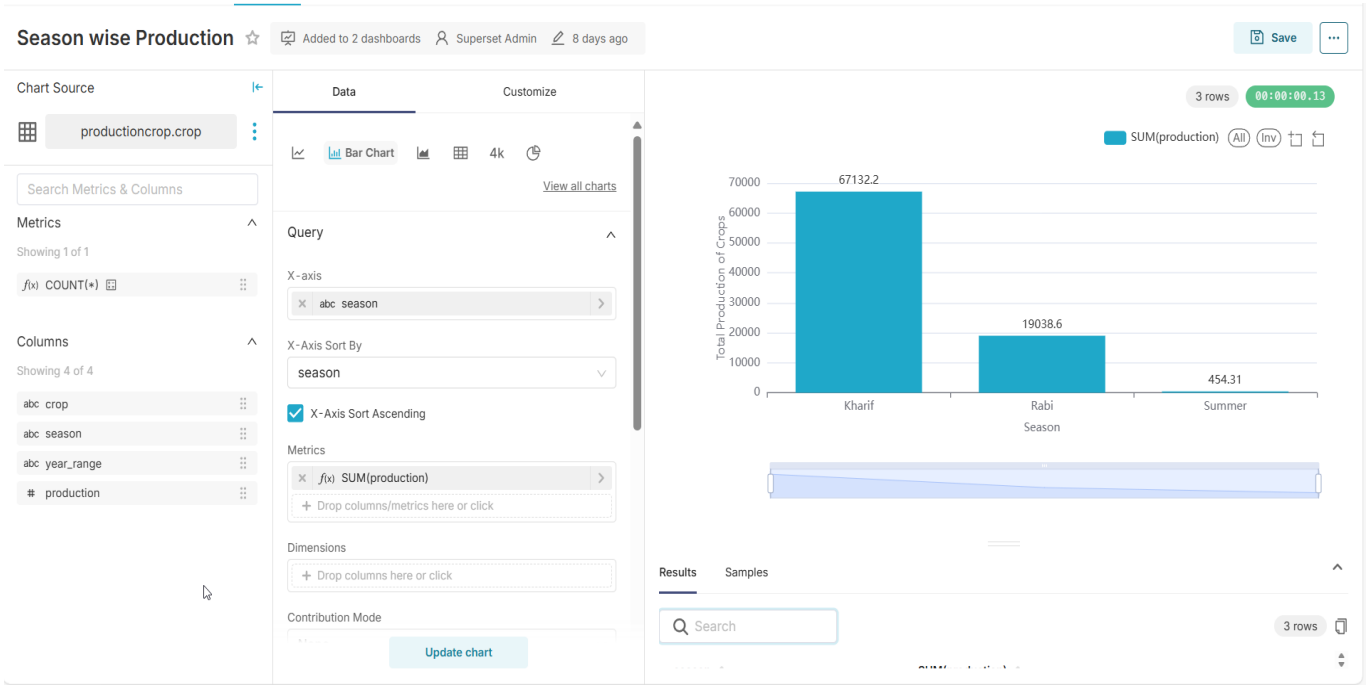


Figure 9: Seasonwise Production

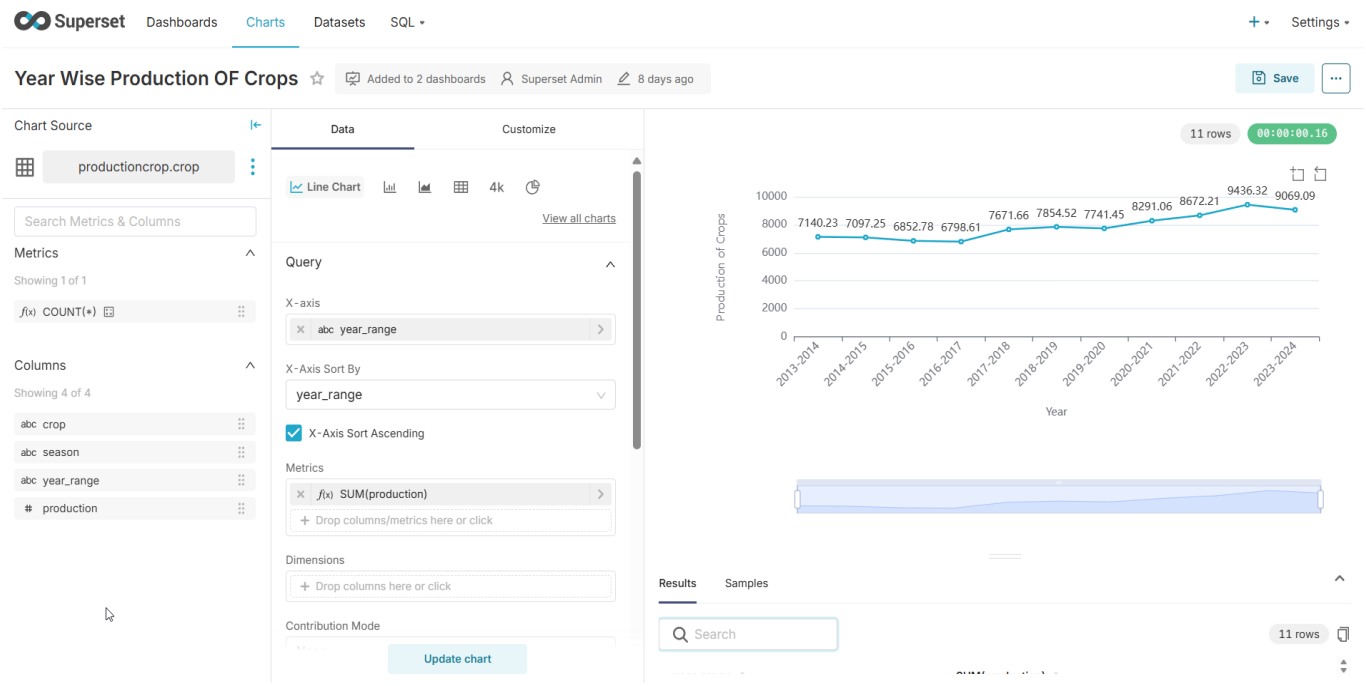


Figure 10: Year Wise Production of Crops

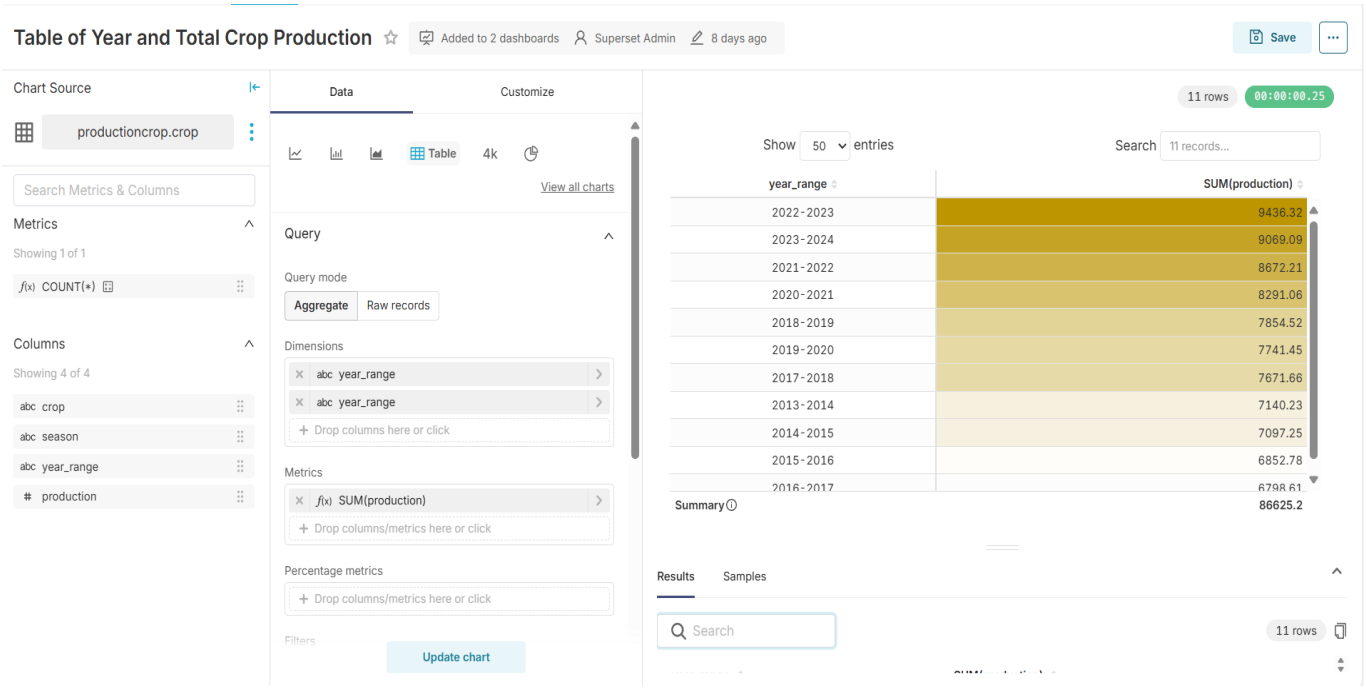


Figure 11: Table of Year and Total Crop Production

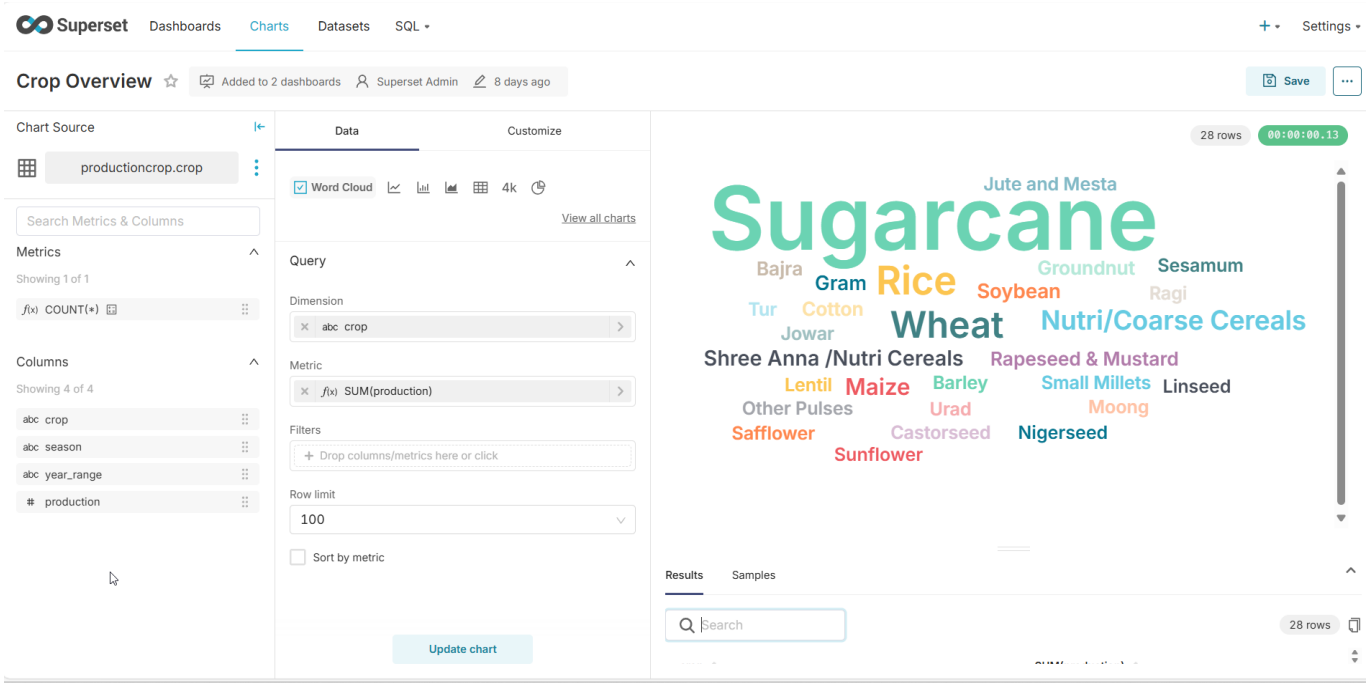


Figure 12: Crop Overview

- **Charts Created in Metabase**

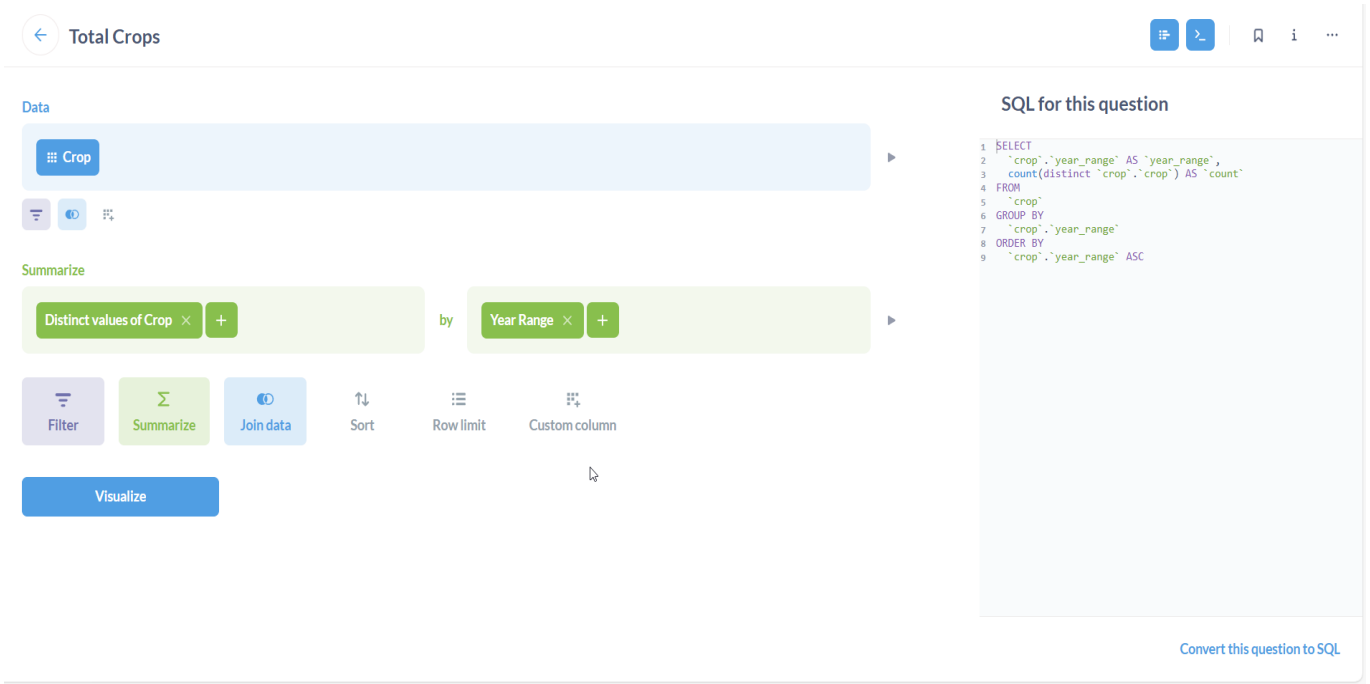


Figure 13: Total Crop

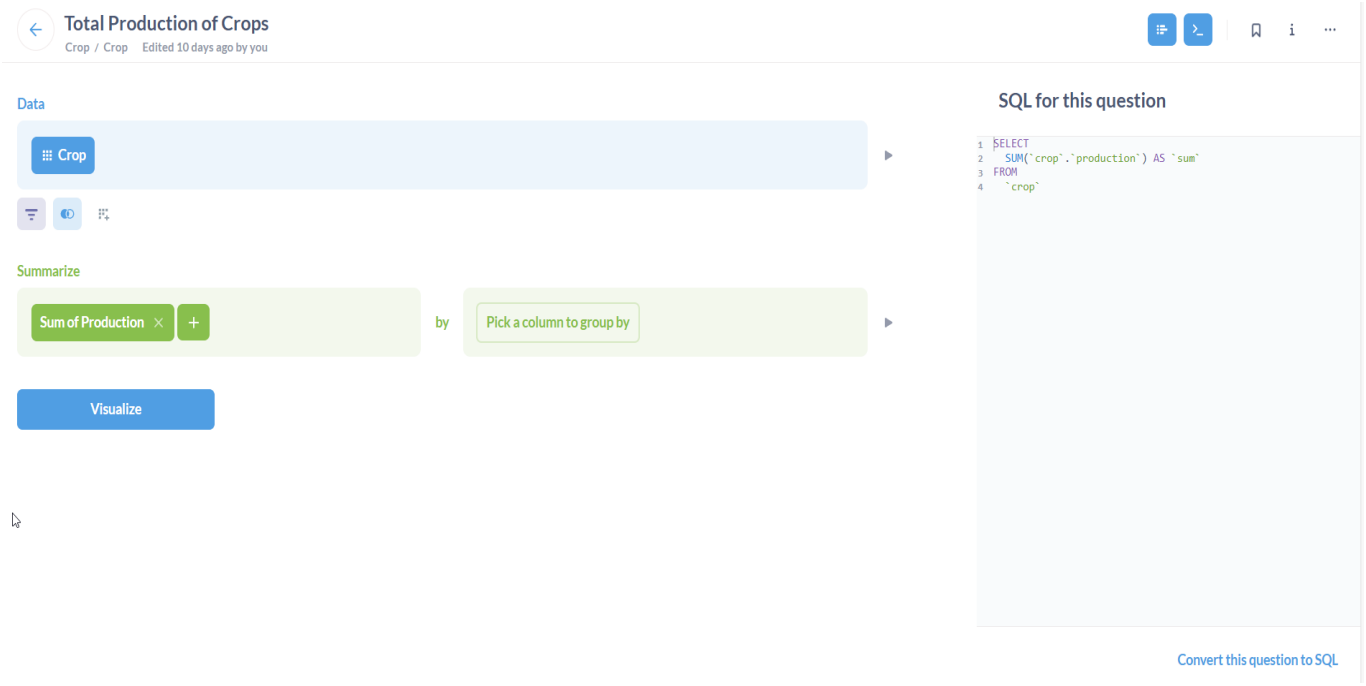


Figure 14: Total Production of Crops (Lakh Tonnes)

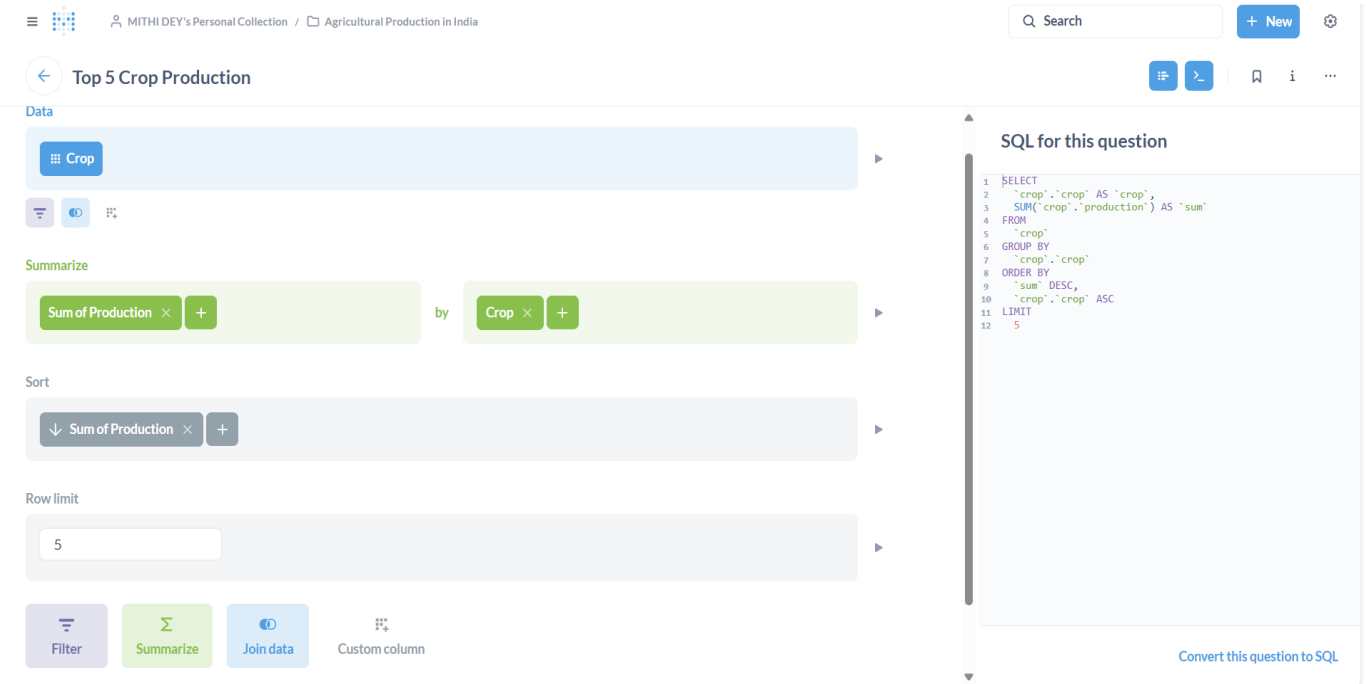


Figure 15: Top 5 Crops

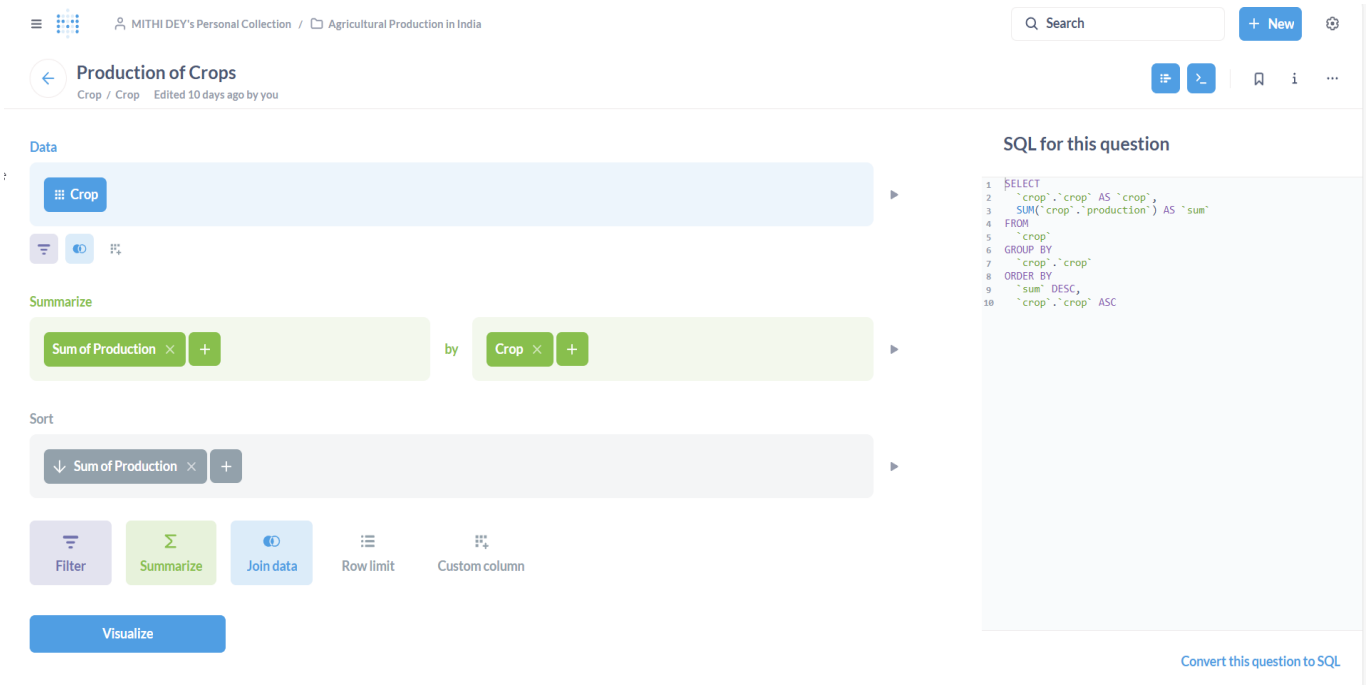


Figure 16: Production of Crops

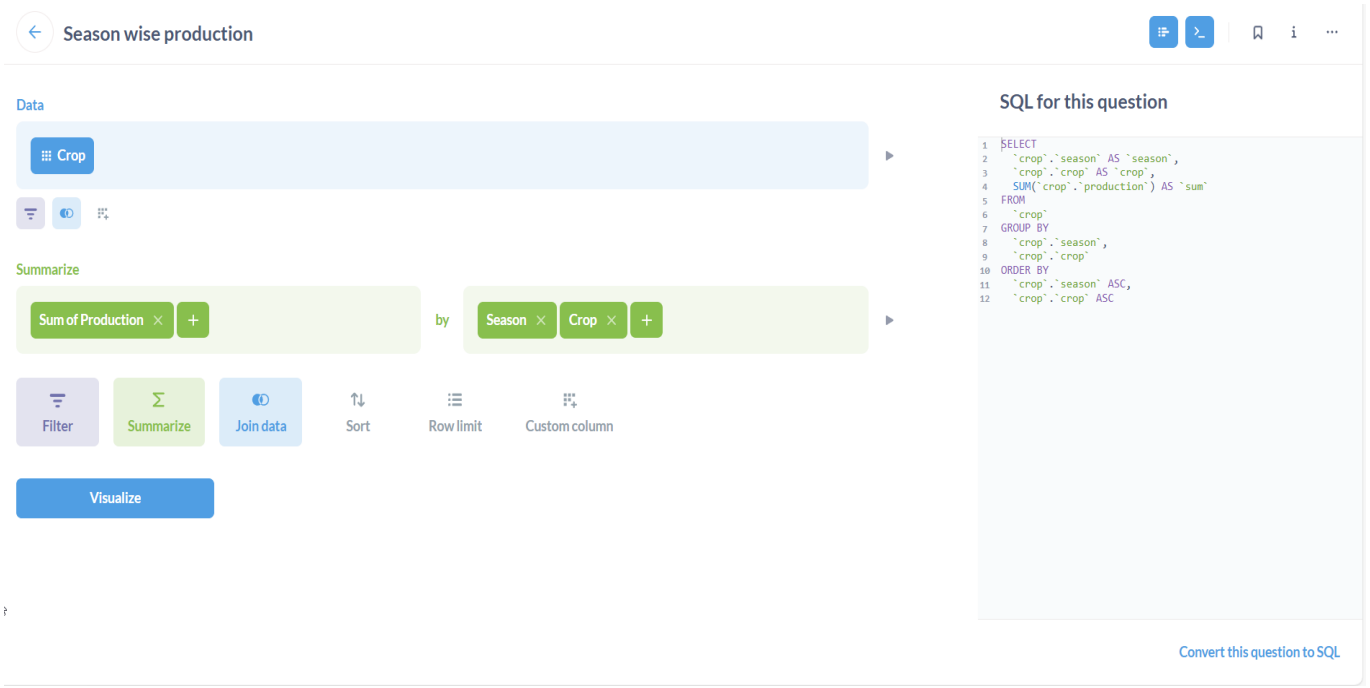


Figure 17: Seasonwise Production

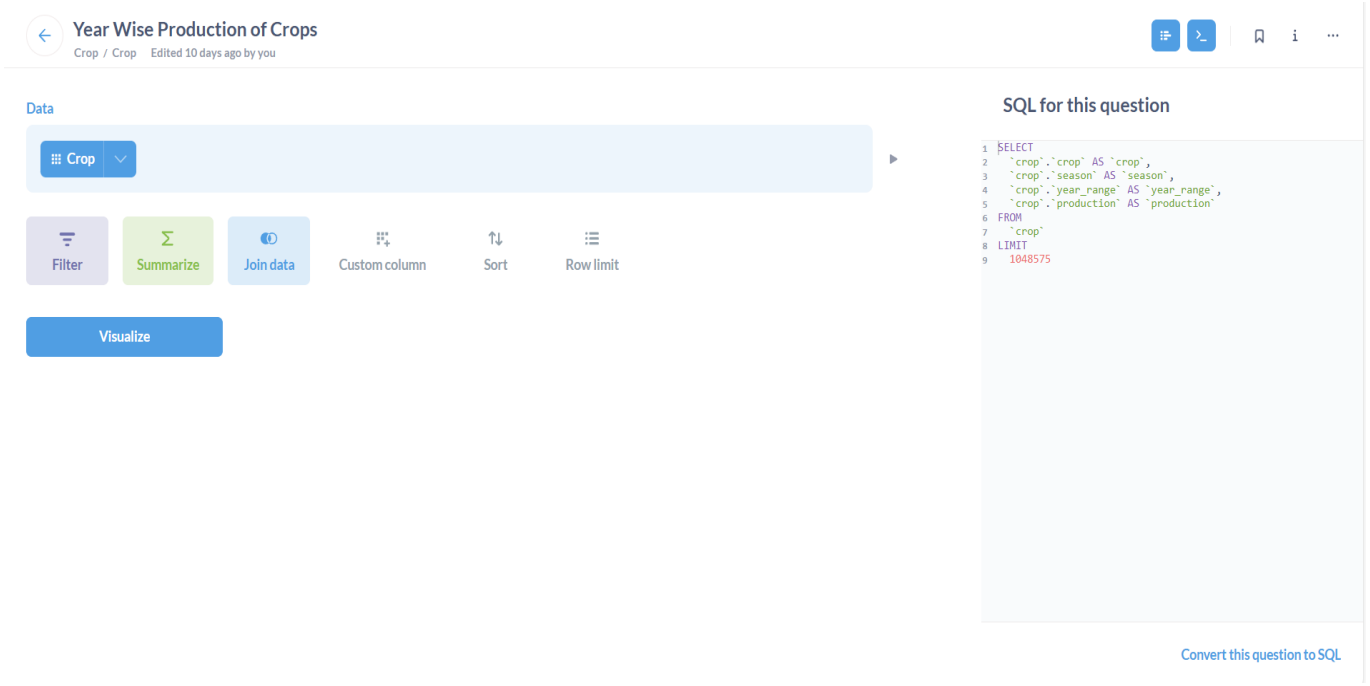


Figure 18: Year Wise Production of Crops

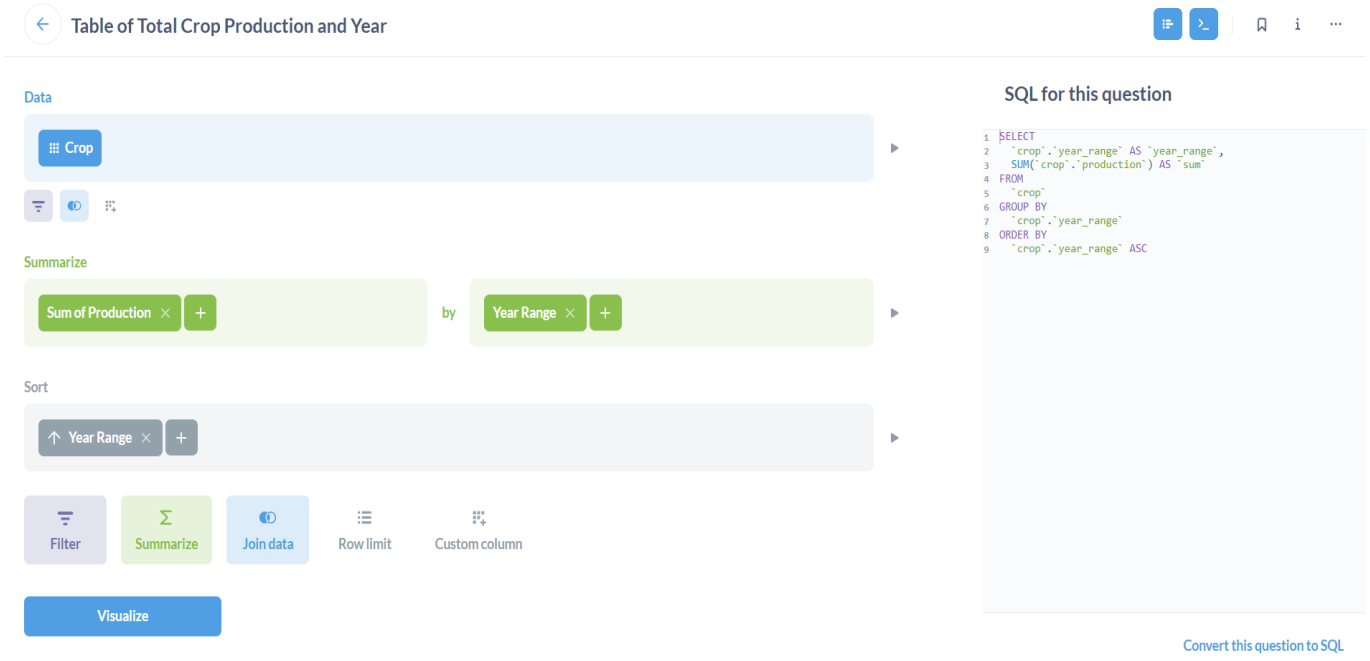


Figure 19: Table of Year and Total Crop Production

Customization was performed on charts in both Metabase and Superset.