Contents

[Requirement 1: Total trip count per month. 1](#_Toc201739303)

[Requirement 2: Total passenger count per month. 2](#_Toc201739304)

[Requirement 3: Average daily income. 2](#_Toc201739305)

[Requirement 4: Tips distribution by passenger count. 3](#_Toc201739306)

[Requirement 5: Average fare per mile. 4](#_Toc201739307)

# Copilot General

**Steps to Use Apache Superset**

**1. Log In**

* Open your browser and go to your Superset URL.
* Enter your username and password to log in.

**2. Connect to a Data Source**

* Go to **Data** > **Databases**.
* Click **+ Database** to add a new connection.
* Choose your database type (e.g., PostgreSQL, MySQL, etc.).
* Enter connection details (host, port, username, password, database name).
* Test the connection and save.

**3. Create a Dataset**

* Go to **Data** > **Datasets**.
* Click **+ Dataset**.
* Select the database and schema, then choose the table you want to explore.
* Save the dataset.

**4. Explore Data**

* Go to **Explore** from the dataset list or click the dataset name.
* Use the drag-and-drop interface to:
  + Choose chart type (bar, line, pie, etc.).
  + Select metrics and dimensions.
  + Apply filters and groupings.
* Click **Run** to generate the chart.

**5. Build a Dashboard**

* Go to **Dashboards** > **+ Dashboard**.
* Name your dashboard and save.
* Click **Edit Dashboard** to add charts.
* Drag and drop charts into the layout.
* Save and publish the dashboard.

**6. Share and Schedule**

* Share dashboards via link or embed code.
* Set up alerts and reports under **Alerts & Reports** to schedule email updates.

**7. Manage Permissions (Optional)**

* Go to **Security** > **List Roles** to manage user access.
* Assign roles and permissions as needed.

# Gemini

As a seasoned data engineer, here's a step-by-step guide to achieving each of the data visualization requirements in Apache Superset, assuming you have the necessary data loaded and accessible in Superset as a dataset (e.g., a trips dataset containing trip information with columns like trip\_id, passenger\_count, fare\_amount, tip\_amount, trip\_distance, and a timestamp column like pickup\_datetime).

For each visualization, the general flow will be:

1. **Create/Select Dataset:** Ensure you have the relevant dataset ready.
2. **Create New Chart:** Start building a new chart.
3. **Configure Chart:** Select chart type, define metrics, dimensions, and other properties.
4. **Save Chart:** Give it a meaningful name.
5. **Add to Dashboard:** Place the chart on your desired dashboard.

Let's break down each requirement:

## Requirement 1: Total trip count per month.

**Goal:** Show the number of trips grouped by month.

**Step-by-step Guidelines:**

1. **Navigate to Charts:**
   * From the Superset home page, click the **"+"** sign in the top right corner and select **"Chart."**
2. **Choose Dataset:**
   * In the "Choose a dataset" dialog, select your relevant dataset (e.g., trips). Click **"Create new chart."**
3. **Configure Chart:**
   * **Visualization Type:** Select **"Bar Chart"** or **"Line Chart"** (Bar Chart is often clearer for counts per period).
   * **Time Column:**
     + Locate the time column (e.g., pickup\_datetime).
     + For "Time Grain," select **"month"**.
   * **Metrics:**
     + The default "COUNT(\*)" or "row\_count" metric should suffice. If not, add a new metric:
       - Click **"Add Metric."**
       - Set "Label" to Total Trips.
       - Set "Aggregate" to COUNT.
       - Set "Column" to trip\_id (or any non-null column that represents a single trip).
       - Click **"Save."**
   * **X-axis:** The "Time Column" configured with "month" time grain will automatically serve as your X-axis.
   * **Y-axis:** The "Total Trips" metric will be on the Y-axis.
   * **Run Query:** Click **"Run Query"** to see the chart preview.
4. **Save Chart:**
   * Click **"Save."**
   * **Chart Name:** Enter Total Trip Count per Month.
   * **Add to Dashboard:** Select an existing dashboard or choose **"Create new dashboard"** and provide a name (e.g., Trip Analysis Dashboard).
   * Click **"Save & Go to Dashboard"** (or "Save & Add to Dashboard" if creating new).

## Requirement 2: Total passenger count per month.

**Goal:** Show the sum of passengers across all trips, grouped by month.

**Step-by-step Guidelines:**

1. **Navigate to Charts:**
   * Click the **"+"** sign and select **"Chart."**
2. **Choose Dataset:**
   * Select the trips dataset. Click **"Create new chart."**
3. **Configure Chart:**
   * **Visualization Type:** Select **"Bar Chart"** or **"Line Chart."**
   * **Time Column:**
     + Locate the time column (e.g., pickup\_datetime).
     + For "Time Grain," select **"month"**.
   * **Metrics:**
     + Click **"Add Metric."**
     + Set "Label" to Total Passengers.
     + Set "Aggregate" to SUM.
     + Set "Column" to passenger\_count.
     + Click **"Save."**
   * **X-axis:** The "Time Column" (monthly grain) will be your X-axis.
   * **Y-axis:** The "Total Passengers" metric will be on the Y-axis.
   * **Run Query:** Click **"Run Query."**
4. **Save Chart:**
   * Click **"Save."**
   * **Chart Name:** Enter Total Passenger Count per Month.
   * **Add to Dashboard:** Select your Trip Analysis Dashboard (or the one you created earlier).
   * Click **"Save & Go to Dashboard"** (or "Save & Add to Dashboard").

## Requirement 3: Average daily income.

**Goal:** Calculate the average daily total income (e.g., fare\_amount + tip\_amount + other charges if applicable), aggregated daily.

**Step-by-step Guidelines:**

1. **Navigate to Charts:**
   * Click the **"+"** sign and select **"Chart."**
2. **Choose Dataset:**
   * Select the trips dataset. Click **"Create new chart."**
3. **Configure Chart:**
   * **Visualization Type:** Select **"Line Chart"** (good for trends over time) or **"Big Number"** (if you want just the grand average). For daily trend, Line Chart is better.
   * **Time Column:**
     + Locate the time column (e.g., pickup\_datetime).
     + For "Time Grain," select **"day"**.
   * **Metrics:**
     + First, you might need a calculated column for Total Income if it doesn't exist. Go to your dataset in "Datasets" tab, click "Edit", then "Calculated Columns" and add a column like fare\_amount + tip\_amount + tolls\_amount (adjust as per your dataset's income components). Let's assume you've named it total\_income.
     + Now back in the chart, add a new metric:
       - Click **"Add Metric."**
       - Set "Label" to Average Daily Income.
       - Set "Aggregate" to AVG.
       - Set "Column" to total\_income (or the expression fare\_amount + tip\_amount if not pre-calculated).
       - Click **"Save."**
   * **X-axis:** The "Time Column" (daily grain) will be your X-axis.
   * **Y-axis:** The "Average Daily Income" metric will be on the Y-axis.
   * **Run Query:** Click **"Run Query."**
4. **Save Chart:**
   * Click **"Save."**
   * **Chart Name:** Enter Average Daily Income.
   * **Add to Dashboard:** Select your Trip Analysis Dashboard.
   * Click **"Save & Go to Dashboard."**

## Requirement 4: Tips distribution by passenger count.

**Goal:** Understand how tip amounts vary based on the number of passengers in a trip.

**Step-by-step Guidelines:**

1. **Navigate to Charts:**
   * Click the **"+"** sign and select **"Chart."**
2. **Choose Dataset:**
   * Select the trips dataset. Click **"Create new chart."**
3. **Configure Chart:**
   * **Visualization Type:** A **"Bar Chart"** (showing average tips per passenger count) or a **"Box Plot"** (showing distribution) would be effective. Let's use Bar Chart for simplicity.
   * **Metrics:**
     + Click **"Add Metric."**
     + Set "Label" to Average Tip Amount.
     + Set "Aggregate" to AVG.
     + Set "Column" to tip\_amount.
     + Click **"Save."**
   * **Group By:**
     + For "Group By," select passenger\_count. This will group the average tip by each distinct passenger count.
   * **Sort By (Optional):** You might want to sort by passenger\_count or Average Tip Amount for better readability.
   * **Run Query:** Click **"Run Query."**
4. **Save Chart:**
   * Click **"Save."**
   * **Chart Name:** Enter Tips Distribution by Passenger Count.
   * **Add to Dashboard:** Select your Trip Analysis Dashboard.
   * Click **"Save & Go to Dashboard."**

## Requirement 5: Average fare per mile.

**Goal:** Calculate the average fare amount divided by the trip distance, to understand the fare efficiency.

**Step-by-step Guidelines:**

1. **Navigate to Charts:**
   * Click the **"+"** sign and select **"Chart."**
2. **Choose Dataset:**
   * Select the trips dataset. Click **"Create new chart."**
3. **Configure Chart:**
   * **Visualization Type:** A **"Big Number"** chart if you want a single average value across all trips, or a **"Bar Chart"** if you want to see this average grouped by another dimension (e.g., payment type, or even just the overall average over time with "Line Chart"). Let's go with "Big Number" for the overall average.
   * **Metrics:**
     + Click **"Add Metric."**
     + Set "Label" to Average Fare per Mile.
     + Set "Aggregate" to AVG.
     + Set "SQL Expression": fare\_amount / trip\_distance
       - **Important Note:** Handle potential trip\_distance being zero to avoid division by zero errors. A more robust expression would be CASE WHEN trip\_distance > 0 THEN fare\_amount / trip\_distance ELSE 0 END or AVG(fare\_amount) / AVG(trip\_distance) depending on your desired aggregation. For simplicity, we'll use fare\_amount / trip\_distance but be aware of data quality.
     + Click **"Save."**
   * **Time Column (Optional for Big Number):** For a single "Big Number," you might not need a time column. If you want to see the trend of "Average fare per mile" over time, you would select pickup\_datetime and a relevant "Time Grain" (e.g., day or month) and use a "Line Chart."
   * **Run Query:** Click **"Run Query."**
4. **Save Chart:**
   * Click **"Save."**
   * **Chart Name:** Enter Average Fare per Mile.
   * **Add to Dashboard:** Select your Trip Analysis Dashboard.
   * Click **"Save & Go to Dashboard."**

After creating all charts, you'll be directed to your Trip Analysis Dashboard. From there, you can **"Edit Dashboard"** to:

* Drag and drop charts to arrange them.
* Resize charts for optimal viewing.
* Add layout elements like rows, columns, and tabs for better organization.
* Set interactive filters if needed (e.g., filter by payment\_type or passenger\_count).
* Finally, **"Save"** your dashboard and **"Publish"** it for broader access.

## Gemini 2 / Refined & More detailed

As a seasoned data engineer, here's an updated and more detailed guide to achieving each of the data visualization requirements in Apache Superset, specifically using the "Yellow Taxi Trip Records" dataset from NYC TLC, and assuming your data is stored in a Hive database accessed via Trino.

We'll break this down into three main sections:

A. Setting up the Data Connection (Hive via Trino): This covers connecting Superset to Trino and exposing your Hive tables.

B. Creating a Dataset in Superset: This explains how to define the dataset Superset will use for visualizations.

C. Creating Each Visualization: Detailed steps for each of your five requirements.

### A. Setting up the Data Connection (Hive via Trino)

This assumes you have a Trino (formerly PrestoSQL) cluster set up and configured to connect to your Hive Metastore.

1. **Log in to Apache Superset:**
   * Open your web browser and navigate to your Superset instance URL.
   * Log in with your credentials (e.g., admin/training).
2. **Add a New Database Connection in Superset:**
   * In the top right corner, click the **"+" (plus) sign** and select **"Data"** then **"Connect Database."**
   * In the "Connect a database" window:
     + **Choose your database:** Select **"Trino"** from the list.
     + **SQLAlchemy URI:** This is the connection string for Trino. It typically looks like: trino://<trino\_host>:<trino\_port>/<catalog>/<schema>?source=<source\_name>
       - **trino\_host**: The hostname or IP address of your Trino coordinator.
       - **trino\_port**: The port Trino is listening on (default is often 8080 or 443 for HTTPS).
       - **<catalog>**: This is your Hive catalog name in Trino (often hive by default, or whatever you configured in your Trino etc/catalog/hive.properties file).
       - **<schema> (Optional but Recommended):** The default schema you want to connect to. For Hive, this would be a database within Hive (e.g., default, taxi\_data, etc.). If omitted, you'll need to specify the schema later in SQL Lab.
       - **source=<source\_name> (Optional):** A descriptive source name for Trino's query tags.
       - **Example URI:** trino://my-trino-coordinator:8080/hive/yellow\_taxi\_db (assuming yellow\_taxi\_db is where your taxi data tables reside in Hive).
     + **Display Name:** Give your connection a user-friendly name (e.g., Trino\_Hive\_Taxi\_Data).
     + **Optional Parameters:** You might need to fill in additional parameters for authentication (e.g., username/password if Trino is secured, or client certificate paths). Consult your Trino setup documentation.
     + **SQL Lab: Default Schema:** Set this to the Hive database where your Yellow Taxi Trip Records tables are located (e.g., yellow\_taxi\_db). This makes querying easier in SQL Lab.
   * Click **"Test Connection"** to ensure Superset can reach Trino. You should get a "Connection looks good!" message.
   * Click **"Connect."**

**B. Creating a Dataset in Superset**

Once your Trino connection is established, you need to create a Dataset in Superset that points to your Hive table. Let's assume your Yellow Taxi Trip Records table is named yellow\_tripdata within the yellow\_taxi\_db Hive database.

1. **Navigate to Datasets:**
   * In Superset, go to **"Data"** > **"Datasets"** from the top menu.
   * Click the **"+ Dataset"** button.
2. **Select Database and Table:**
   * **Database:** Select the Trino connection you just created (e.g., Trino\_Hive\_Taxi\_Data).
   * **Schema:** Enter the Hive database name where your table resides (e.g., yellow\_taxi\_db).
   * **Table Name:** Enter the name of your Yellow Taxi Trip Records table (e.g., yellow\_tripdata).
   * Click **"Add."**
3. **Configure the Dataset (Crucial Step!):**
   * Superset will now show the dataset details. This is where you configure columns, metrics, and ensure data types are correct.
   * **Columns Tab:**
     + Review the columns imported from Hive.
     + **Identify Time Columns:** Find your primary timestamp column (e.g., tpep\_pickup\_datetime or lpep\_pickup\_datetime).
       - For this column, ensure its "Type" is set to "DATETIME."
       - Mark it as "Is temporal" by checking the checkbox.
     + **Numerical Columns:** Ensure fare\_amount, tip\_amount, total\_amount, trip\_distance, passenger\_count are set to appropriate numerical types (e.g., "NUMERIC" or "FLOAT").
     + **Other Columns:** Verify types for other columns.
   * **Metrics Tab:**
     + By default, Superset might create a COUNT(\*) metric. You can add more:
     + **Calculated Column for Total Income:**
       - Click **"Add Metric."**
       - **Label:** Total Income
       - **Expression:** fare\_amount + tip\_amount + tolls\_amount + extra + mta\_tax + improvement\_surcharge (adjust based on actual columns in your dataset that contribute to income).
       - **Aggregate:** SUM (or AVG if you want average per record). Leave as SUM for now.
       - Click **"Save."**
     + **Calculated Column for Fare per Mile (handling zero distance):**
       - Click **"Add Metric."**
       - **Label:** Fare Per Mile
       - **Expression:** CASE WHEN trip\_distance > 0 THEN fare\_amount / trip\_distance ELSE 0 END (This handles cases where trip\_distance might be 0, preventing division-by-zero errors. You can choose to filter out 0-distance trips later if preferred).
       - **Aggregate:** AVG (to get the average fare per mile across trips).
       - Click **"Save."**
   * **Save Changes:** Click **"Save"** at the bottom of the dataset configuration page.

Your dataset is now ready for visualization!

**C. Creating Each Visualization**

Now, let's create the charts based on the requirements, using the yellow\_tripdata dataset.

**Requirement 1: Total trip count per month.**

**Goal:** Number of taxi trips aggregated by month.

**Step-by-step Guidelines:**

1. **Start New Chart:**
   * Go to **"Charts"** from the top menu, then click **"+ Chart."**
2. **Choose Dataset:**
   * Select your yellow\_tripdata dataset. Click **"Create new chart."**
3. **Configure Chart:**
   * **Visualization Type:** Select **"Line Chart"** (good for time series trends) or **"Bar Chart"**.
   * **Time Column:**
     + **Time Column:** Select tpep\_pickup\_datetime (or your main pickup timestamp column).
     + **Time Grain:** Choose **"month"**.
   * **Metrics:**
     + **Metric:** Select COUNT(\*) (default row\_count if present and accurate, otherwise create a new metric as COUNT(trip\_id)).
   * **Run Query:** Click **"Run Query"** to see the preview.
4. **Save Chart:**
   * Click **"Save."**
   * **Chart Name:** Total Trip Count per Month
   * **Add to Dashboard:** Select Create new dashboard and name it Yellow Taxi Data Dashboard. Click **"Save & Go to Dashboard."**

**Requirement 2: Total passenger count per month.**

**Goal:** Sum of passengers across all trips, aggregated by month.

**Step-by-step Guidelines:**

1. **Start New Chart:**
   * Go to **"Charts"** > **"+ Chart."**
2. **Choose Dataset:**
   * Select yellow\_tripdata. Click **"Create new chart."**
3. **Configure Chart:**
   * **Visualization Type:** Select **"Line Chart"** or **"Bar Chart"**.
   * **Time Column:**
     + **Time Column:** Select tpep\_pickup\_datetime.
     + **Time Grain:** Choose **"month"**.
   * **Metrics:**
     + **Metric:** Click **"Add Metric."**
       - **Label:** Total Passengers
       - **Aggregate:** SUM
       - **Column:** passenger\_count
       - Click **"Save."**
   * **Run Query:** Click **"Run Query."**
4. **Save Chart:**
   * Click **"Save."**
   * **Chart Name:** Total Passenger Count per Month
   * **Add to Dashboard:** Select Yellow Taxi Data Dashboard. Click **"Save & Go to Dashboard."**

**Requirement 3: Average daily income.**

**Goal:** Average of the calculated Total Income (fare + tips + extras) per day.

**Step-by-step Guidelines:**

1. **Start New Chart:**
   * Go to **"Charts"** > **"+ Chart."**
2. **Choose Dataset:**
   * Select yellow\_tripdata. Click **"Create new chart."**
3. **Configure Chart:**
   * **Visualization Type:** Select **"Line Chart"** (to show trend) or **"Big Number"** (for a single overall average). For a daily trend, Line Chart is better.
   * **Time Column:**
     + **Time Column:** Select tpep\_pickup\_datetime.
     + **Time Grain:** Choose **"day"**.
   * **Metrics:**
     + **Metric:** Select the Total Income metric you created in the dataset configuration.
       - If you set its aggregate to SUM in the dataset, here you will change the aggregate to AVG within the chart's "Metrics" section. Superset allows overriding the dataset's default aggregate for a metric here.
   * **Run Query:** Click **"Run Query."**
4. **Save Chart:**
   * Click **"Save."**
   * **Chart Name:** Average Daily Income
   * **Add to Dashboard:** Select Yellow Taxi Data Dashboard. Click **"Save & Go to Dashboard."**

**Requirement 4: Tips distribution by passenger count.**

**Goal:** How average tips vary based on the number of passengers.

**Step-by-step Guidelines:**

1. **Start New Chart:**
   * Go to **"Charts"** > **"+ Chart."**
2. **Choose Dataset:**
   * Select yellow\_tripdata. Click **"Create new chart."**
3. **Configure Chart:**
   * **Visualization Type:** Select **"Bar Chart"** or **"Box Plot"** (Box Plot is excellent for showing distribution, outliers, and quartiles).
   * **Metrics:**
     + **Metric:** Click **"Add Metric."**
       - **Label:** Average Tip
       - **Aggregate:** AVG
       - **Column:** tip\_amount
       - Click **"Save."**
   * **Group By:**
     + **Group By:** Select passenger\_count.
   * **Filters (Optional):** You might want to filter out passenger\_count = 0 if it's not relevant.
     + Click **"Add Filter."**
     + Select passenger\_count as column.
     + Select > (greater than) as operator.
     + Enter 0 as value.
   * **Run Query:** Click **"Run Query."**
4. **Save Chart:**
   * Click **"Save."**
   * **Chart Name:** Tips Distribution by Passenger Count
   * **Add to Dashboard:** Select Yellow Taxi Data Dashboard. Click **"Save & Go to Dashboard."**

**Requirement 5: Average fare per mile.**

**Goal:** Calculate the average fare\_amount divided by trip\_distance for all trips.

**Step-by-step Guidelines:**

1. **Start New Chart:**
   * Go to **"Charts"** > **"+ Chart."**
2. **Choose Dataset:**
   * Select yellow\_tripdata. Click **"Create new chart."**
3. **Configure Chart:**
   * **Visualization Type:** Select **"Big Number"** (for a single overall average) if you want to see just one value. If you want to see how this changes over time, use a **"Line Chart"** and configure a time column.
   * **Metrics:**
     + **Metric:** Select the Fare Per Mile metric you created in the dataset configuration.
   * **Run Query:** Click **"Run Query."**
4. **Save Chart:**
   * Click **"Save."**
   * **Chart Name:** Average Fare per Mile
   * **Add to Dashboard:** Select Yellow Taxi Data Dashboard. Click **"Save & Go to Dashboard."**

**Final Steps: Dashboard Organization and Publishing**

Once all charts are created and added to your Yellow Taxi Data Dashboard:

1. **Go to Dashboard:**
   * Navigate to **"Dashboards"** from the top menu and select your Yellow Taxi Data Dashboard.
2. **Edit Dashboard:**
   * Click the **"Edit Dashboard"** button.
3. **Arrange and Customize:**
   * **Drag and Drop:** Move your charts around to create a logical flow.
   * **Resize:** Adjust the size of each chart to fit your layout.
   * **Layout:** Use the "Add Tab," "Add Row," "Add Column" options to structure your dashboard into sections.
   * **Filters:** Consider adding dashboard-level filters (e.g., date range filter for pickup\_datetime, or a dropdown for payment\_type) for interactive exploration.
4. **Save and Publish:**
   * Click **"Save"** to save your layout changes.
   * Click **"Publish"** to make the dashboard visible and shareable to other users.

This detailed breakdown should provide a clear path to achieving all your visualization requirements in Apache Superset using your Yellow Taxi Trip Records dataset via Trino and Hive. Remember to adapt column names and calculation logic to your specific dataset's schema.