# [https://avatars2.githubusercontent.com/u/4156894?v=3&s=100](http://www.calstatela.edu/centers/hipic)CIS3200 Term Project Tutorial

#### Authors: Roxana Cuevas;  Jenny Iglesias; Dante Villalobos; Abrar Hasan

#### Instructor: [Jongwook Woo](https://www.linkedin.com/in/jongwook-woo-7081a85)

#### Date: 05/17/2021

**Lab Tutorial**

Roxana Cuevas ([rcueva12@calstatela.edu](mailto:rcueva12@calstatela.edu)), Abrar Hasan ([ataha@calstatela.edu](mailto:ataha@calstatela.edu)), Jenny Iglesias ([jiglesi8@calstatela.edu](mailto:jiglesi8@calstatela.edu)), Dante Villalobos(dvilla38[@calstatela.edu](mailto:yourname@calstatela.edu))

05/17/2021

**Air Traffic Data Analysis using Elasticsearch (U.S. International & Domestic Air Traffic Data)**

**Objectives**

**List what your objectives are.** In this hands-on lab, you will learn how to:

* Import a csv data file into elasticsearch
* Create visualizations based on the imported data
* Create a predictive analysis using Azure Machine Learning

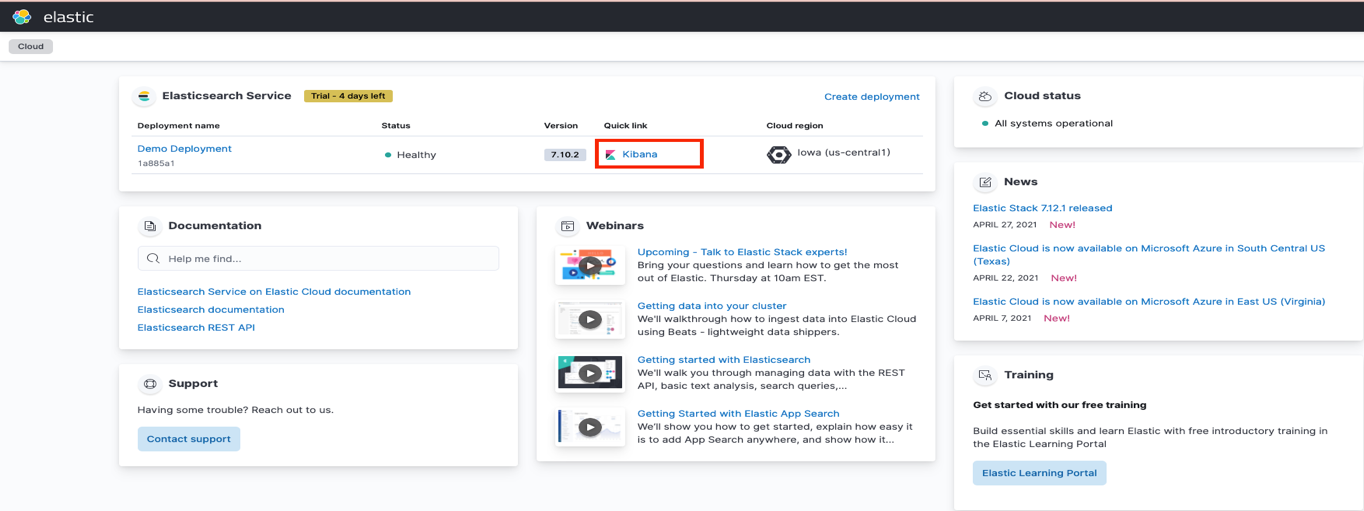
**Platform Specs**

* Processor: Intel Core i5
* CPU Speed: 2.3 GHz
* # of CPU cores: 4
* Total Memory Size: 8GB RAM

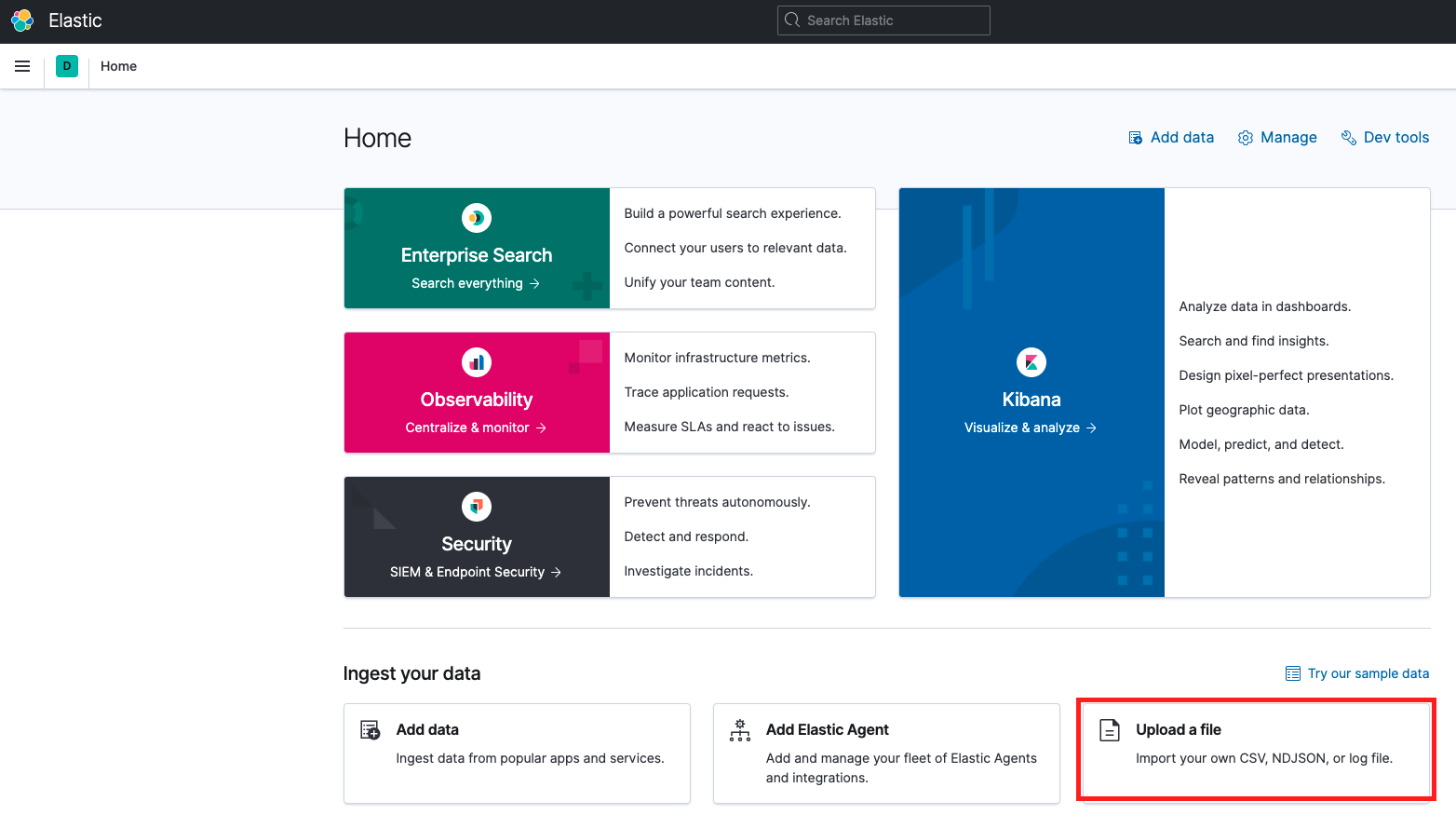
Step 1: Import Data into Kibana

This step is to manually import a csv file already saved on your computer containing data into Kibana.

1. Login to elasticsearch. Go to [https://www.elastic.cocloud/as-a-service](https://www.elastic.co/cloud/as-a-service)
2. Log into your ES (Elastic Cloud) account
3. You will see the following page once logged in. Click on Kibana



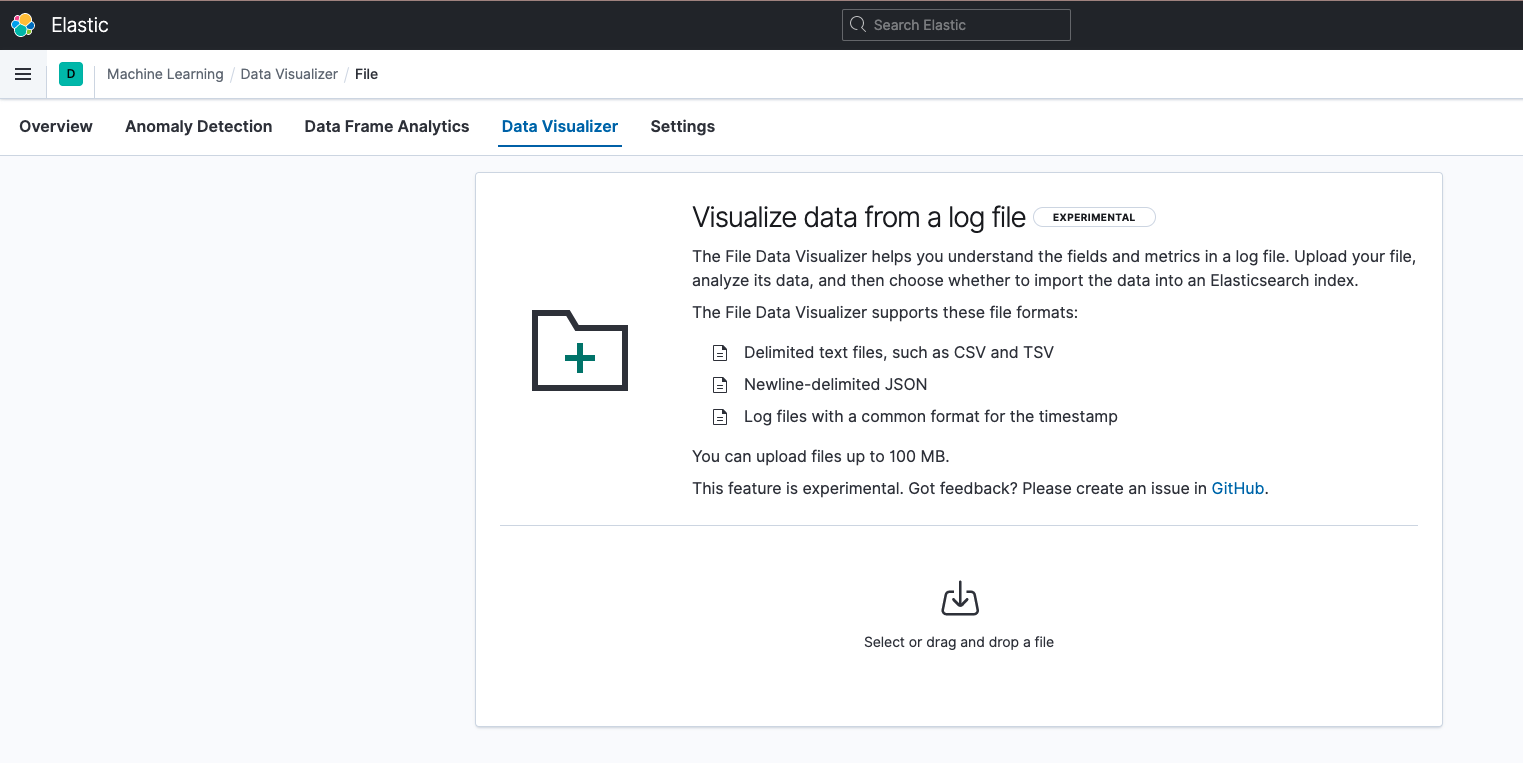
1. You will be directed to the following page. Click upload a file



Ensure you have previously downloaded and saved to your computer a csv file containing data you wish to visualize. We have previously downloaded a file containing air traffic data from

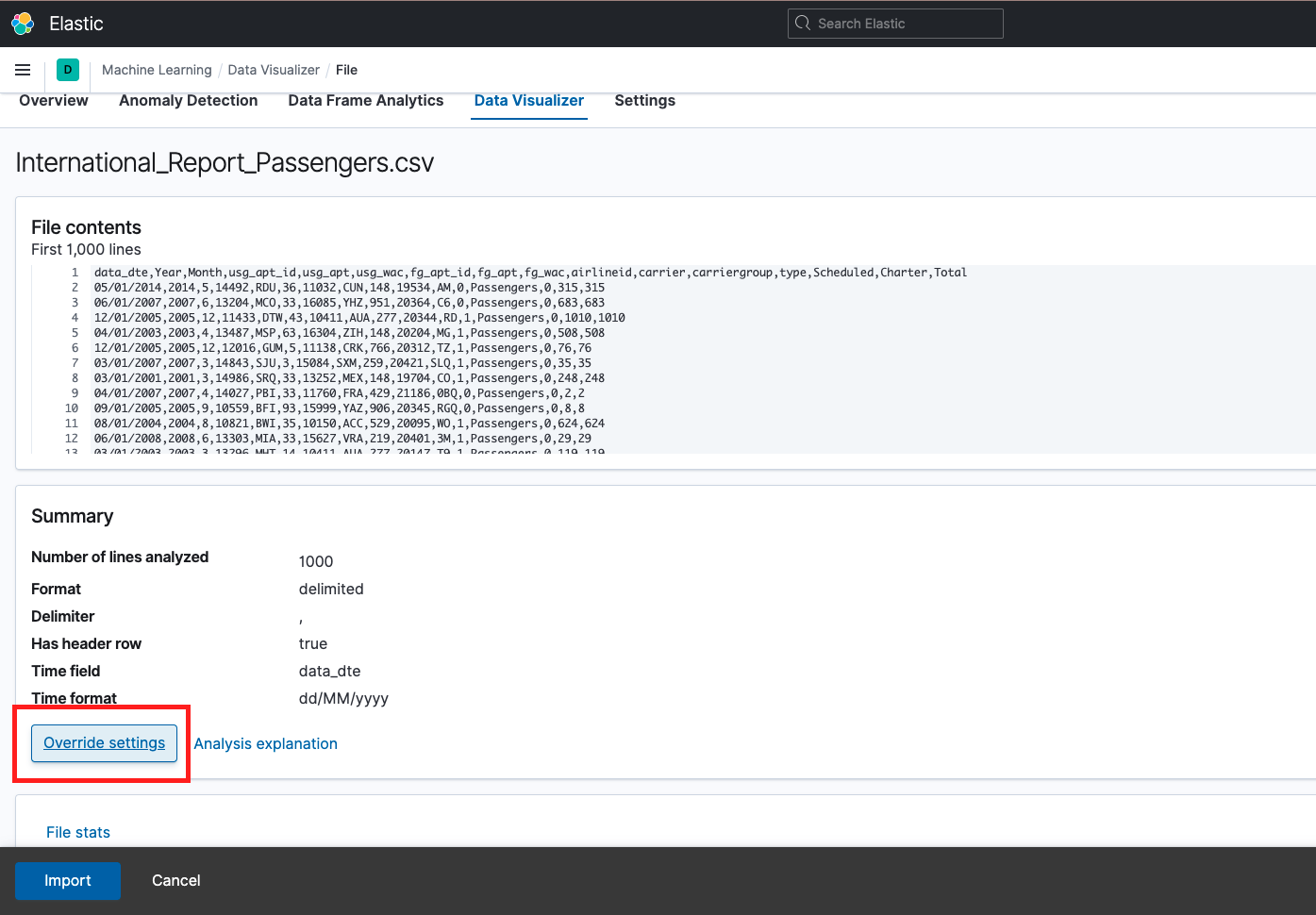
<https://www.kaggle.com/parulpandey/us-international-air-traffic-data>

1. Then click select or drag and drop a file. Ensure you locate the file within your computer directory to select and import or drag and drop.

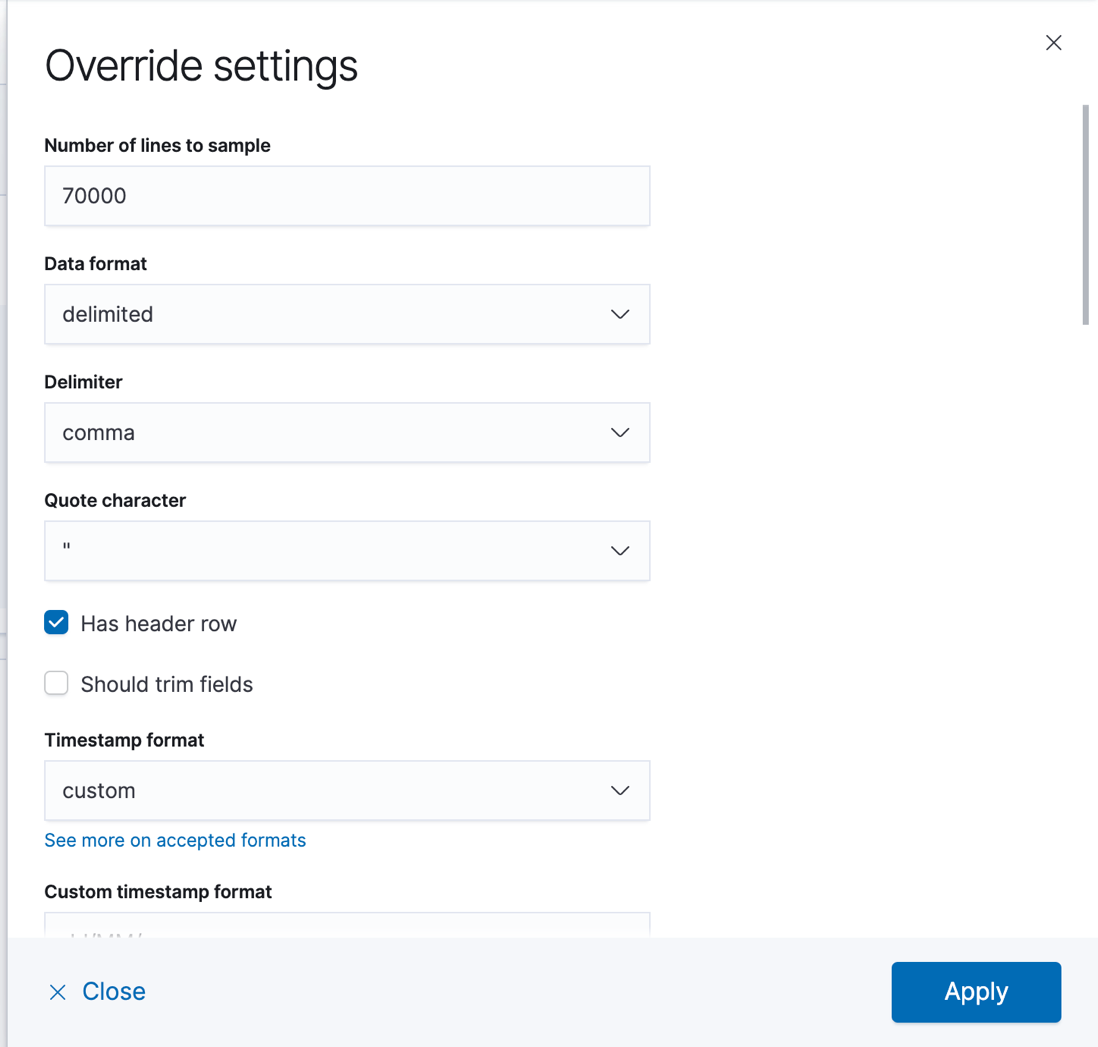


Depending on the size of your dataset you may need to update the number of analyzed lines. The default as you can see below is 1000 lines of analyzed data. However, we want to look at all lines within our dataset. If your dataset contains more than this number you will need to update to consider all lines.

1. To update, click override settings and change the number of lines to sample to 70000. Click Apply.



1. Ensure when you override settings to include 70000 rows that your override setting looks as follows: Data format: delimited, Delimiter : comma, Has header row is checked. Then click Apply



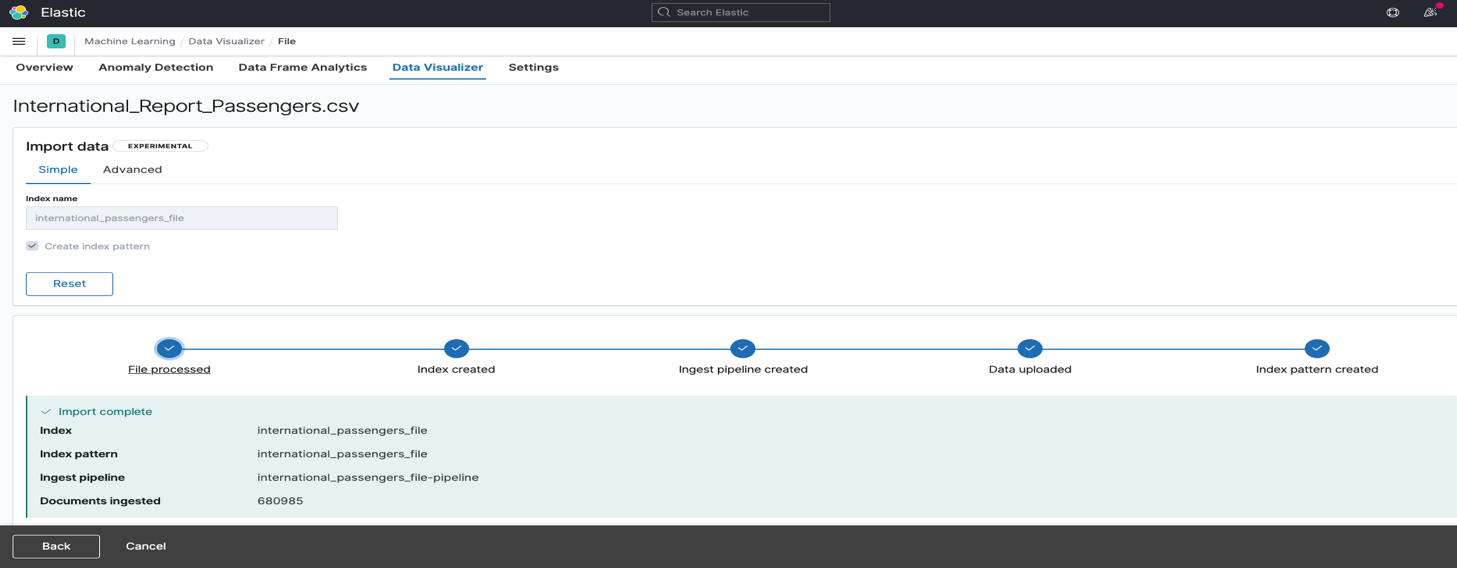
1. Now, all lines within the dataset are included and your screen should look as follows. Click import.



1. Add an index name for the data you are uploading. Then click Import.



1. Data will now be analyzed and imported. Once analysis is complete all checkmarks below will appear, and status will show as below green and import complete.

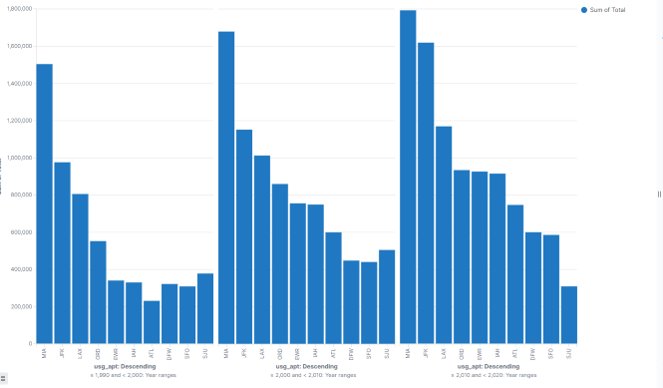


1. Now we are ready to create visualizations and reports.

Step 2: Create Visualizations

This step is to show a detailed step by step of how to create bar graphs, pie charts, line charts and aero spatial maps that are useful in visualizing and better understanding your data.

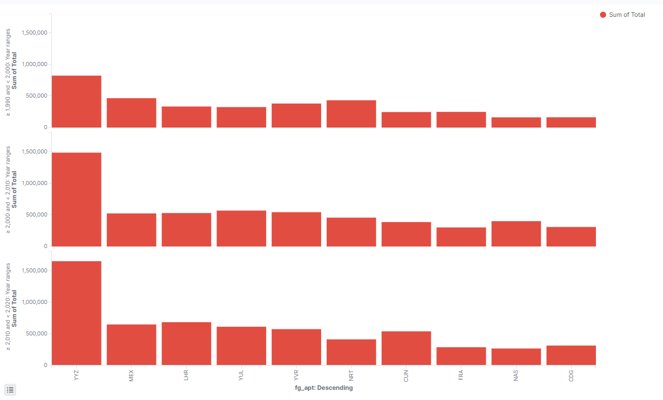
1. Vertical Bar Chart for most domestic flights flying in the United States.
2. In the create visualize section, we selected the dataset and picked vertical bar charts as our visualization.
3. When we got to the creating the visualization section, we had to set the time range from 1990 – 2020.
4. Then we set the y axis to the sum of total flights.
5. We set the x-axis to Year then set three different ranges. The ranges of the years were 1990-2000, 2000-2010,2010-2020.
6. Then we selected the option to split the chart. We set the field to total usg\_apt, which is the domestic airports in the United States. This chart shows which airports had most flights flying in the U.S. from 1990 – 2020. Your chart will look as follows:



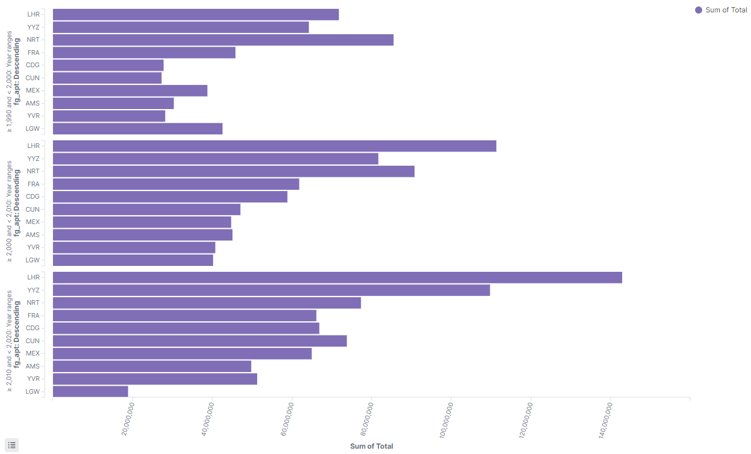
1. Horizontal Bar Chart for most passengers flying in the United States.
2. In the create visualize section, we selected the dataset and picked horizontal bar charts as our visualization.
3. When we got to the creating the visualization section, we had to set the time range from 1990 – 2020.
4. Then we set the x axis to the sum of total passengers.
5. We set the y-axis to Year then set three different ranges. The ranges of the years were 1990-2000, 2000-2010,2010-2020.
6. Then we selected the option to split the chart. We set the field to total usg\_apt, which is the domestic airports in the United States. This chart shows which airports had most passengers flying in the U.S. from 1990 – 2020. Your chart will look as follows:



1. Vertical Bar Chart for most flights flying from international airport in the world.
2. In the create visualize section, we selected the dataset and picked vertical bar charts as our visualization.
3. When we got to the creating the visualization section, we had to set the time range from 1990 – 2020.
4. Then we set the y axis to the sum of total flights.
5. We set the x-axis to Year then set three different ranges. The ranges of the years were 1990-2000, 2000-2010,2010-2020.
6. Then we selected the option to split the chart. We set the field to total fg\_apt, which is the international airports around the world. This chart shows which airports had most flights flying around the world from 1990 – 2020. Your chart will look as follows:



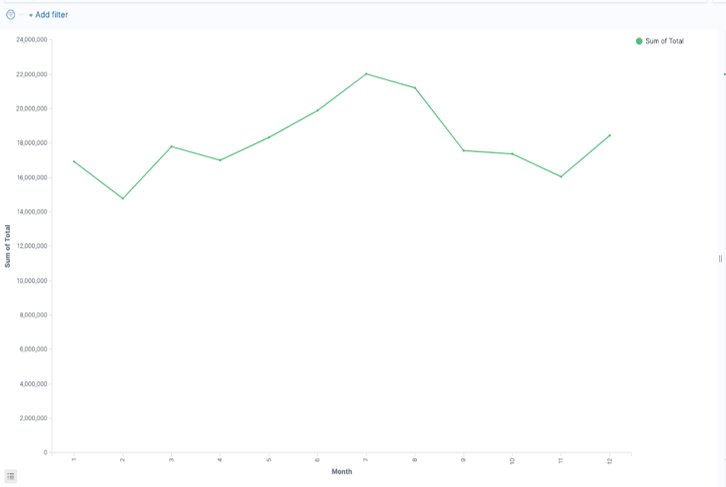
1. Horizontal Bar Chart for most passengers flying from international airports in the world.
2. In the create visualize section, we selected the dataset and picked horizontal bar charts as our visualization.
3. When we got to the creating the visualization section, we had to set the time range from 1990 – 2020.
4. Then we set the x axis to the sum of total passengers.
5. We set the y-axis to Year then set three different ranges. The ranges of the years were 1990-2000, 2000-2010,2010-2020.
6. Then we selected the option to split the chart. We set the field to total fg\_apt, which is the international airports around the world. This chart shows which airports had most passengers flying in the world from 1990 – 2020. Your chart will look as follows:



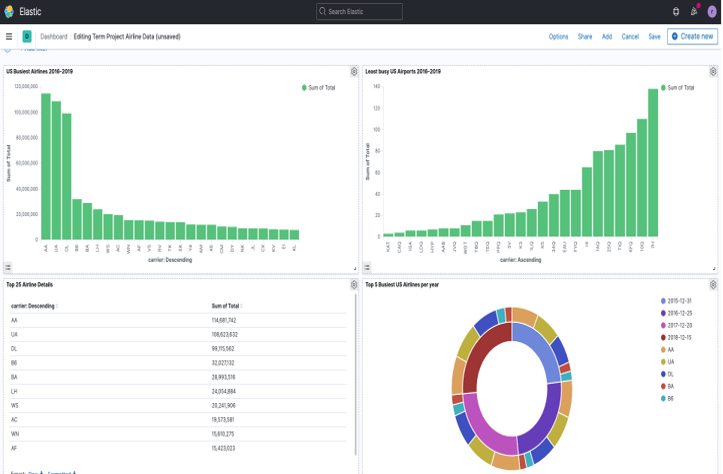
1. For the following visualization we will repeat the same steps 4 times in order to analyze the number of passengers traveling each month. This will allow us to see which are the busiest and least busy months to travel.
2. First in Kibana choose visualize
3. Then click create visualization
4. Click “line”
5. Choose the data set “International\_Reports\_Passengers.csv”
6. Set the date ranges to absolute
7. Set the date ranges to absolute January 1, 2016 00:00:00 to December 31, 2016 00:00:00
8. Under Buckets choose Y axis Sum of Total
9. Add aditional buckets and choose

* Aggregation: Histogram
* Field: Month
* Click update

1. When finished repeat steps e-h and only change the date ranges to Jan 1, 2017-Dec 31, 2017, then 2018 and finally for 2019.
2. When completed, your chart will look as follows:



1. Next you will create a Dashboard, but you must first create the visualizations that will be included in the Dashboard. Once finished, your Dashboard will look as follows:

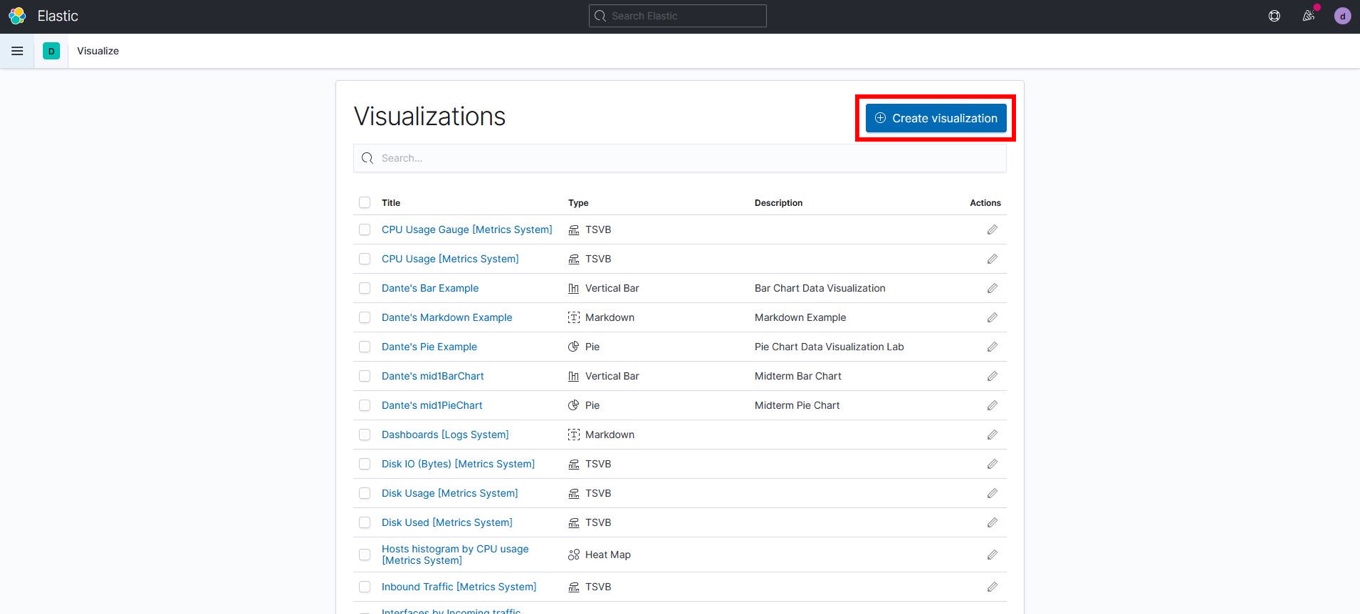


**US Busiest Airlines 2016-2019 Bar Graph**

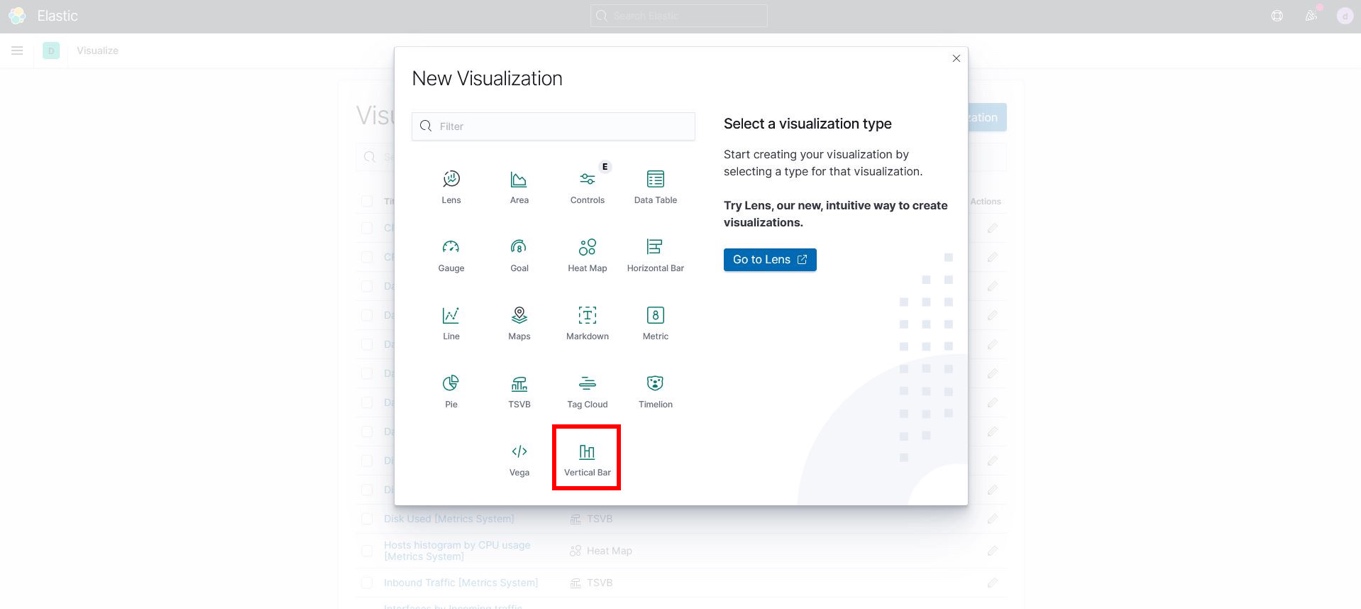
1. On the top left menu choose “Visualize” under Kibana.



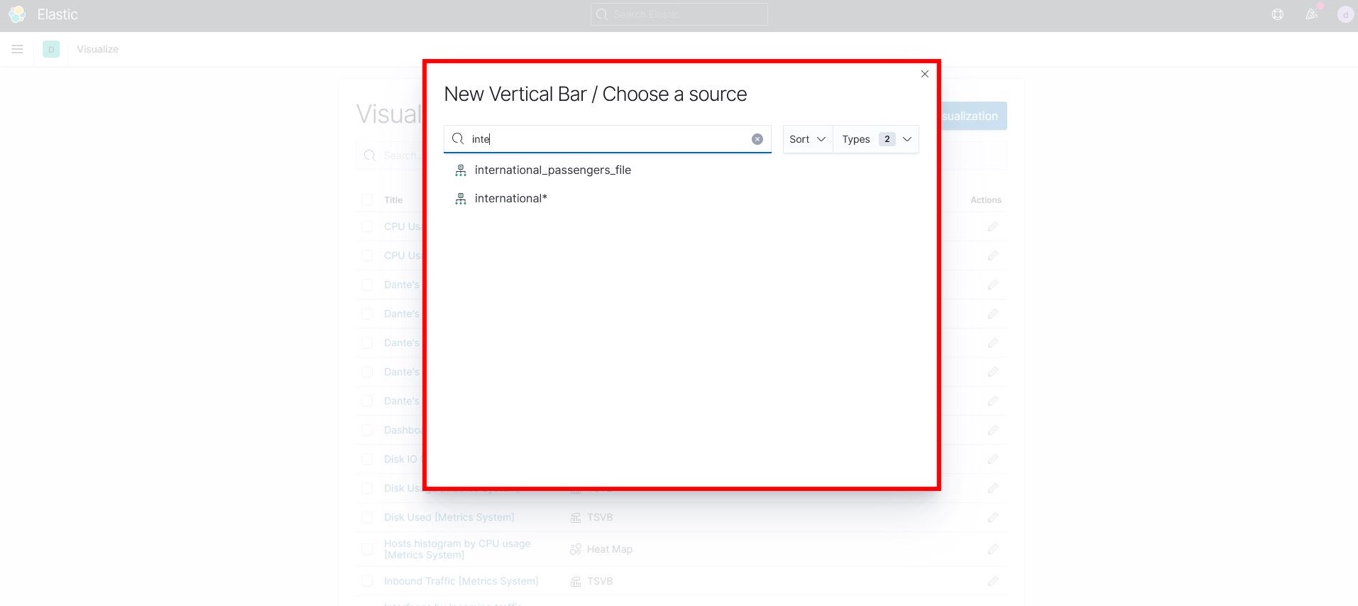
1. Click on “Create Visualization.”



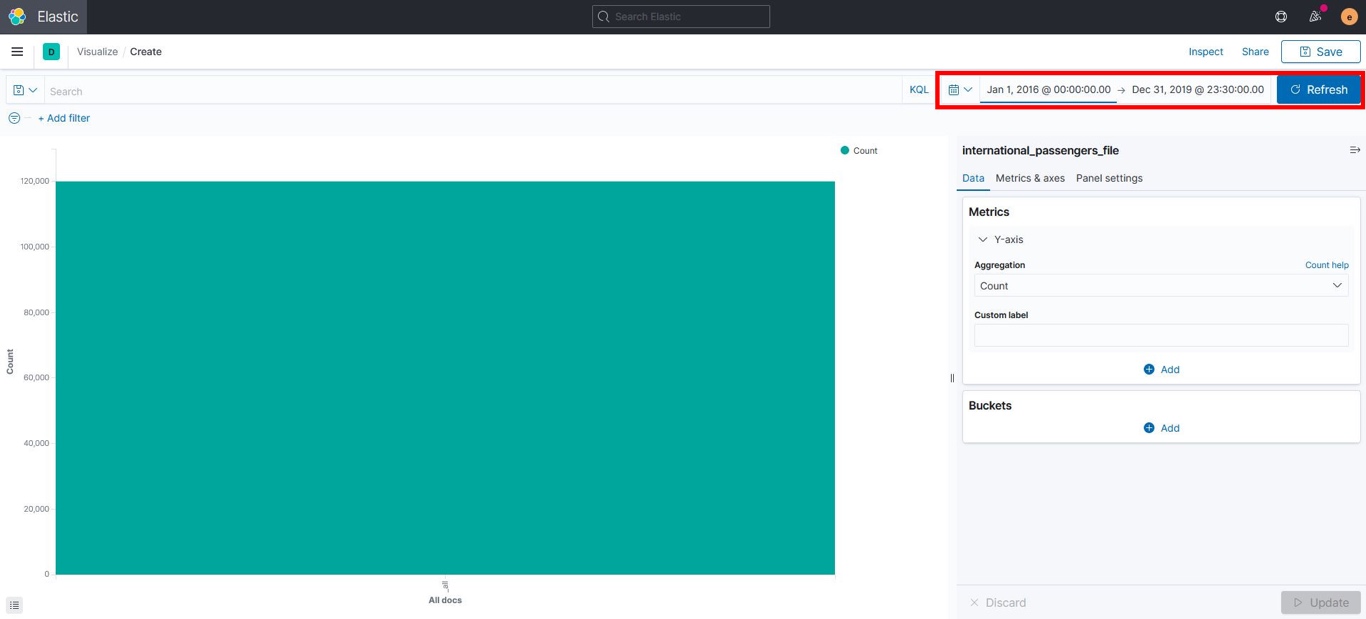
1. Select “Vertical Bar” as the visualization type.



1. Begin typing “int” and select “international\_passengers\_file.”



1. On the top right corner, set the dates as “Absolute” from January 1, 2016 @ 00:00:00 to December 31, 2019 @ 23:30:00. Click Refresh.

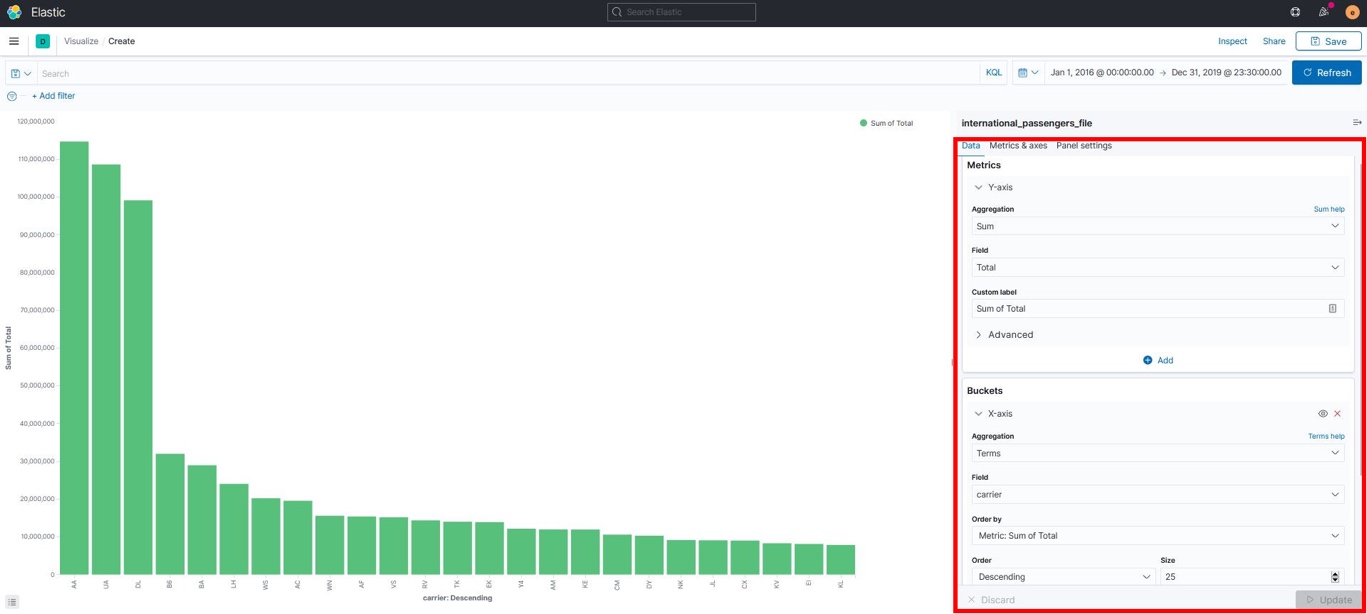


1. In the “Metrics” pane, expand Y-axis Count and use the following settings:

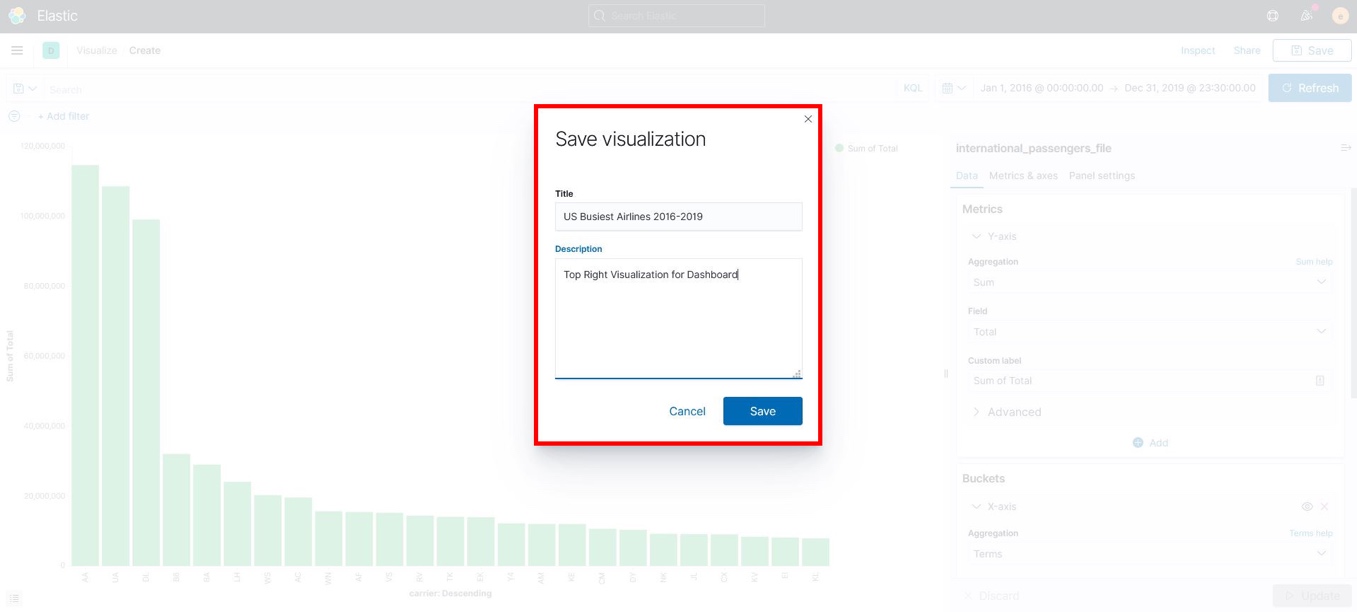
* Aggregation: Sum
* Field: Total
* Custom Label: Sum of Total

Click “Update.

1. In the “Buckets” pane click “Add” and under ADD BUCKET select X-axis and use the following settings:
   * Aggregation: Terms
   * Fields: Carrier
   * Order by: Metric: Sum of Total
   * Order: Descending
   * Size: 25

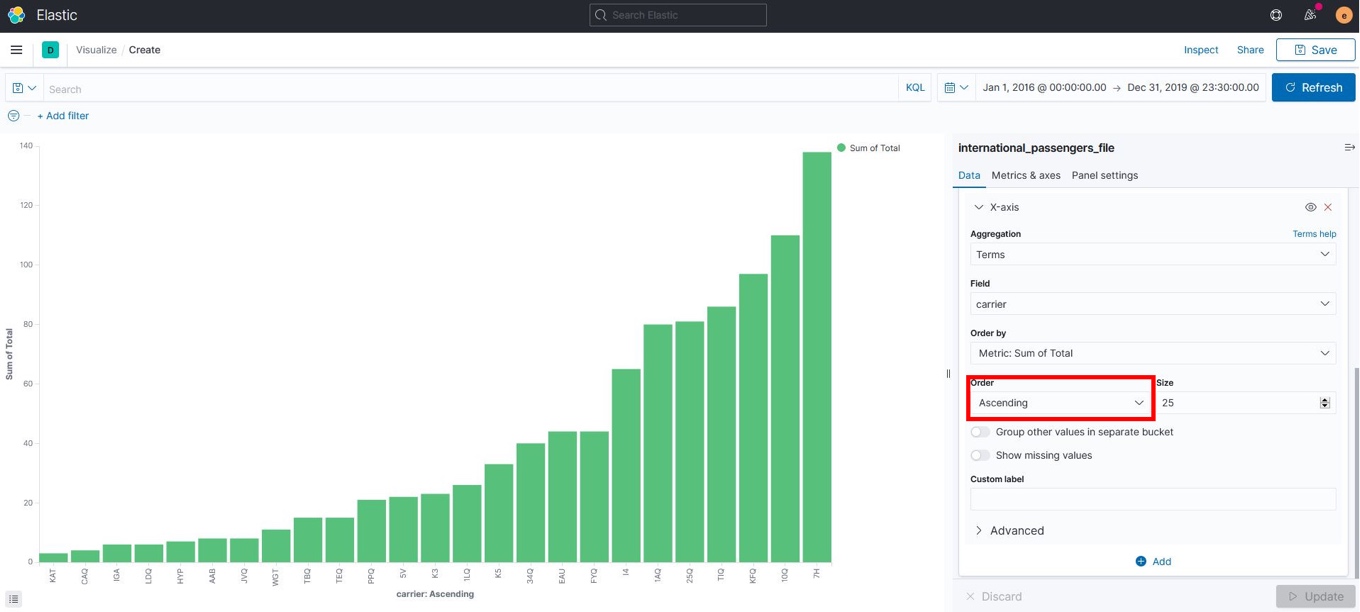


1. Click “Update,” then save your graph and give it a Title and Description.



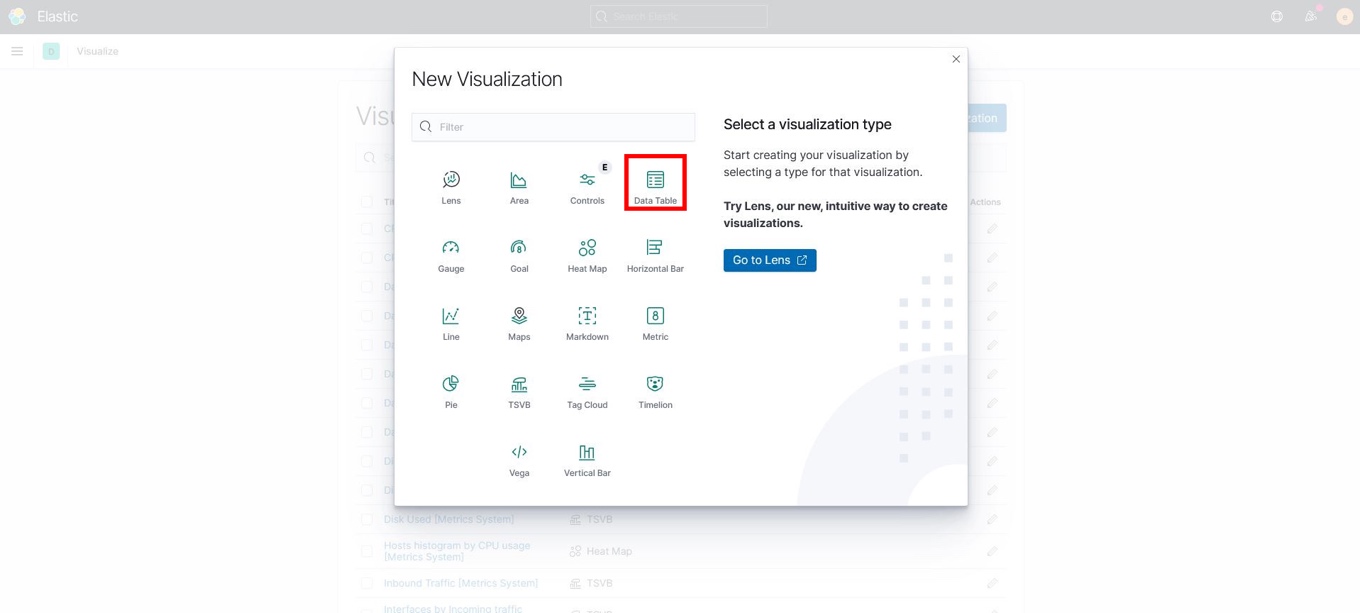
**Least Busy US Airlines 2016-2019 Bar Graph**

1. For this graph you will use the same information from the bar graph you just created. All you have to do is change the Order from Descending to Ascending in the “Buckets” pane as shown in the image below. Save your graph and add a Title and Description.

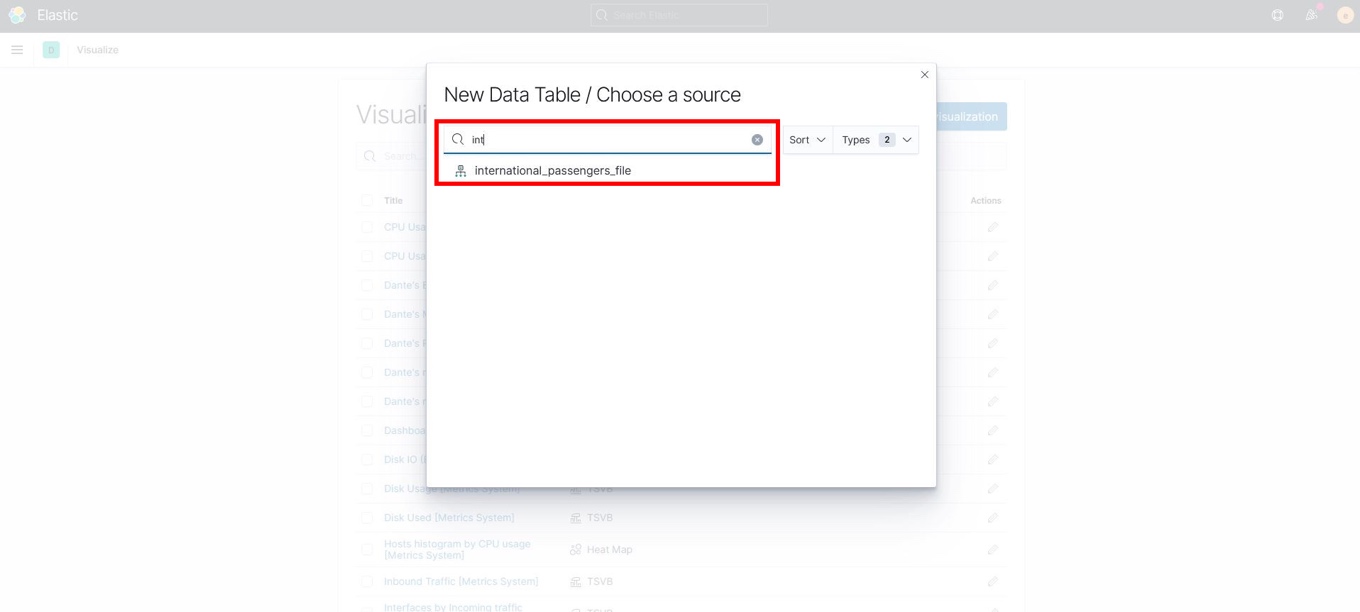


**Top 25 Airlines Details Data Table**

1. In the “Visualize” section, click on “Create Visualization” and select “Data Table.”



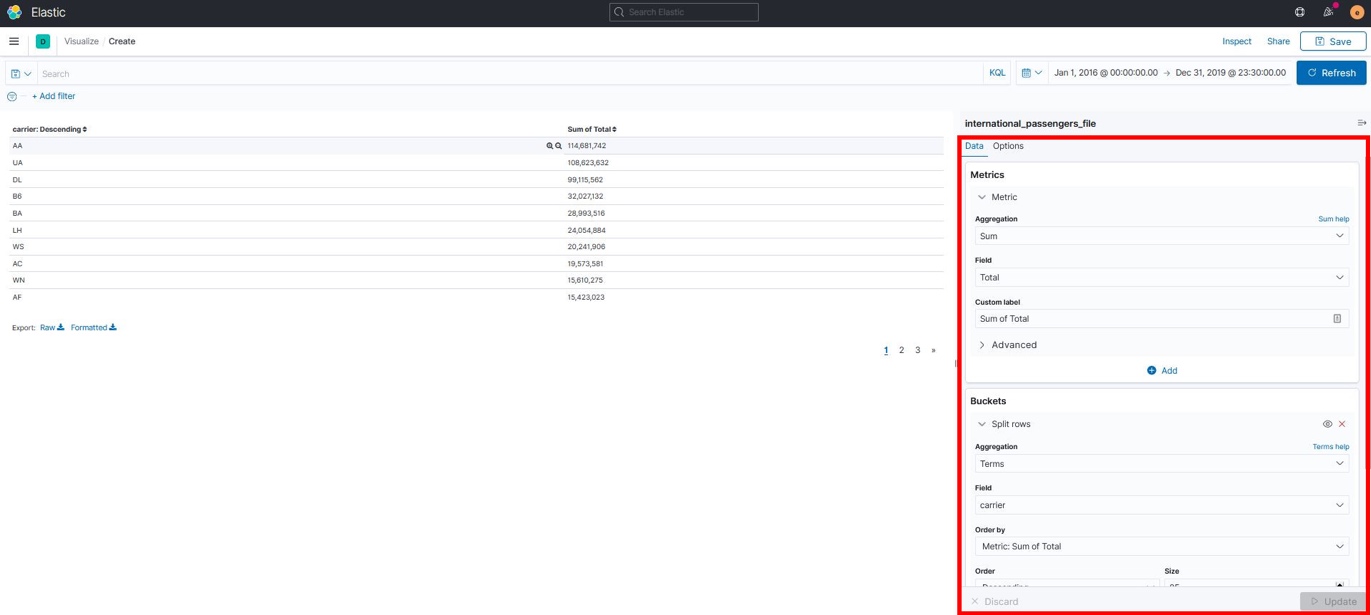
1. Search for “international\_passengers\_file” and select it as the source.



1. In the “Metrics” pane, expand “Metric Count” and use the following settings:

* Aggregation: Sum
* Field: Total
* Custom Label: Sum of Total

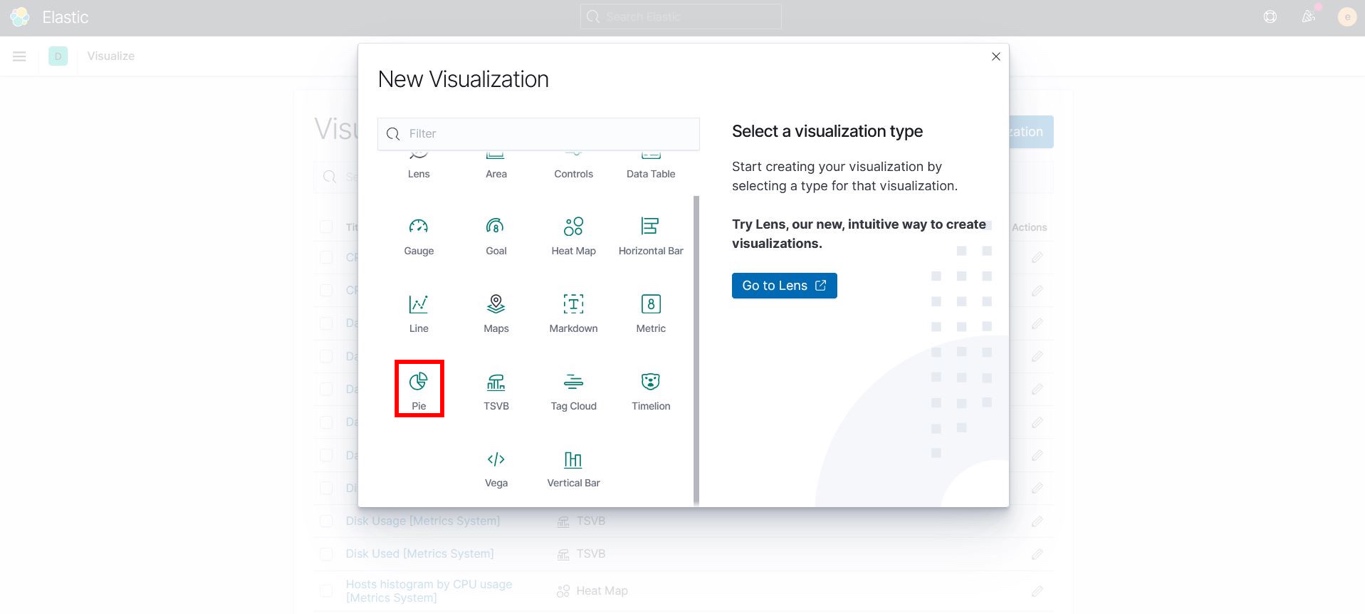
1. In the “Buckets” pane, click “Add” and under ADD BUCKET select “Split Rows,” then use the following settings:
   * Aggregation: Terms
   * Field: carrier
   * Order by: Metric: Sum of Total
   * Order: Descending
   * Size: 25



1. Save your table, then add a Title and Description.

**Top 5 Busiest US Airlines Per Year Pie Chart**

1. In the “Visualize” section, click “Create Visualization” and select “Pie.”

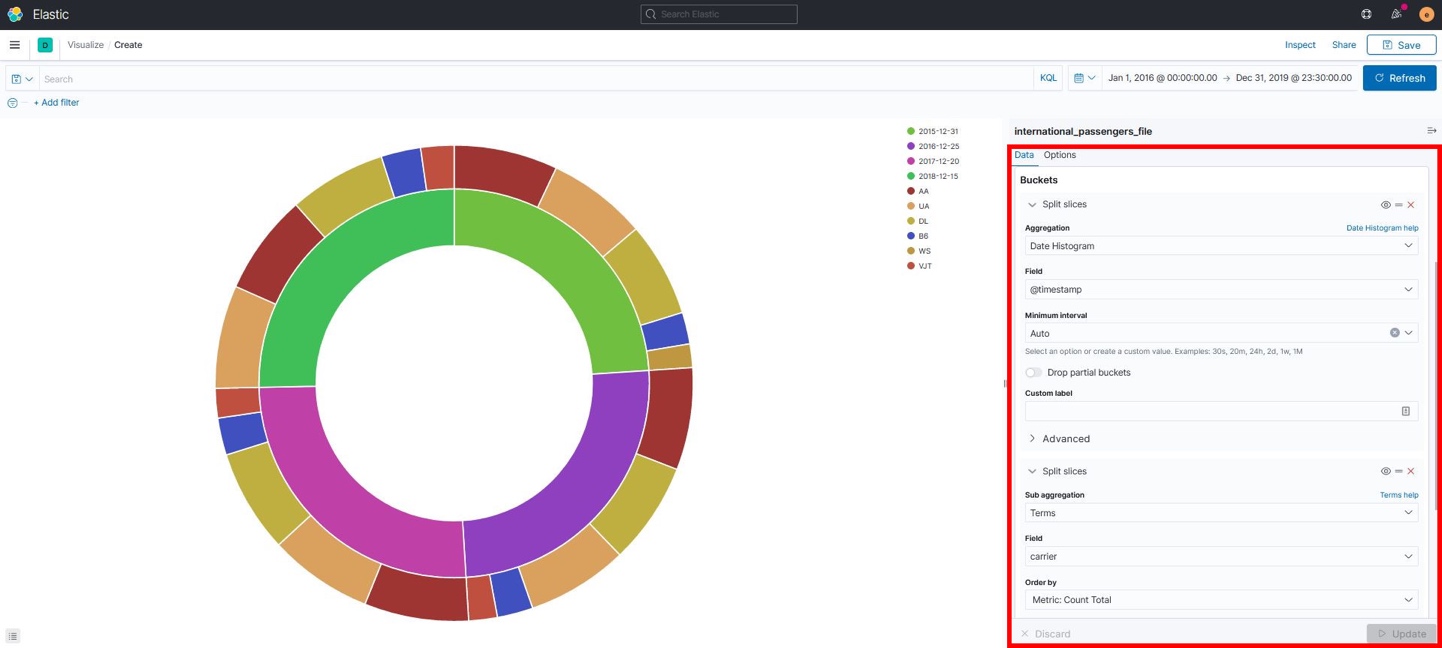


1. Search for “international\_passengers\_file” and select it, as shown in previous steps.
2. In the “Metrics” pane, use the following settings:
   * Aggregation: Count
   * Custom Label: Count Total
3. In the “Buckets” pane, click Add and under ADD BUCKET select “Split Slices.” Use the following settings:

* Aggregation: Date Histogram
* Field: @timestamp
* Minimum Interval: Auto

1. At the bottom of the “Buckets” pane click Add, and under ADD SUB-BUCKET selecct “Split Slices.” Use the following settings for the sub-bucket:

* Sub Aggregation: Terms
* Field: carrier
* Order by: Metric: Count Total
* Order: Descending
* Size: 5

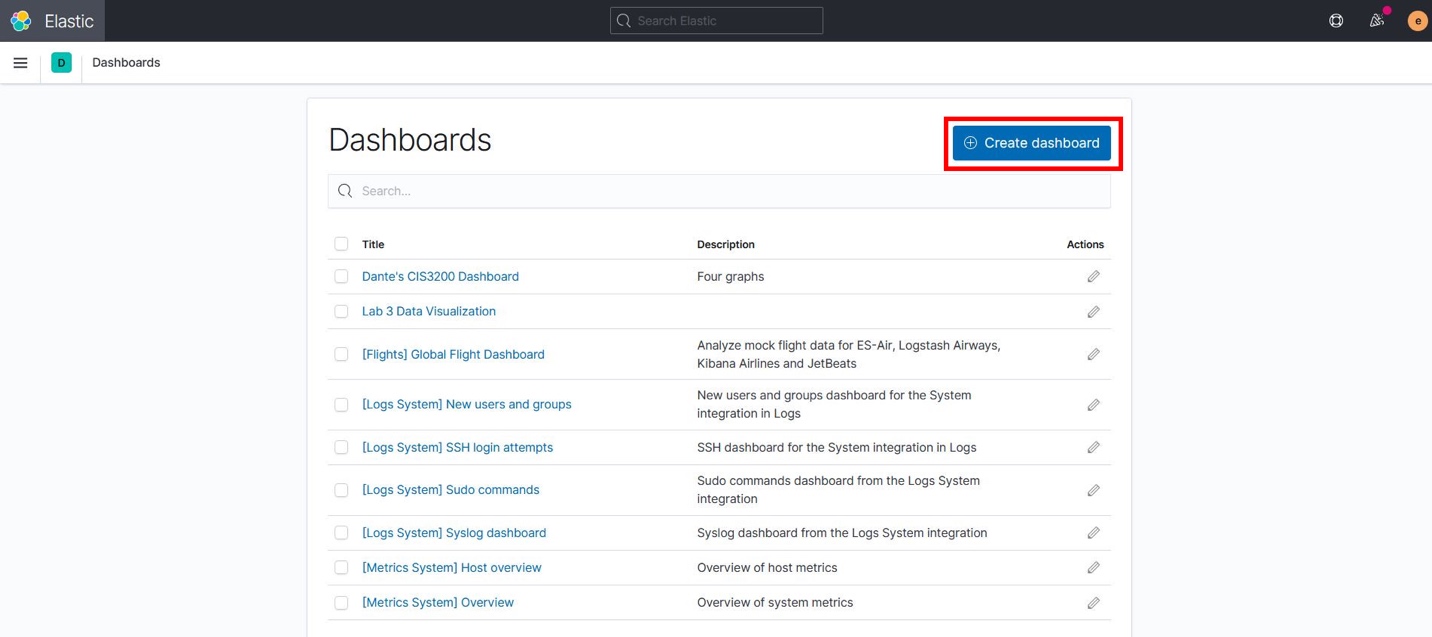
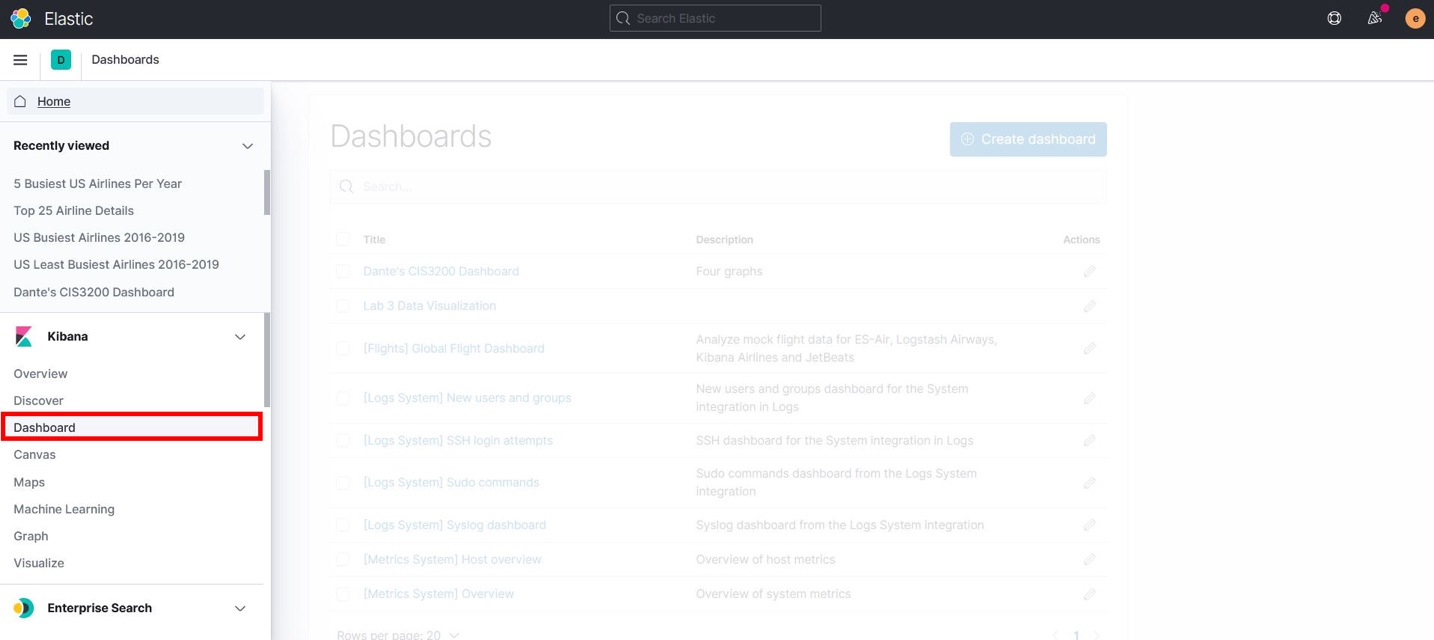


1. Save your chart, then add a Title and Description.

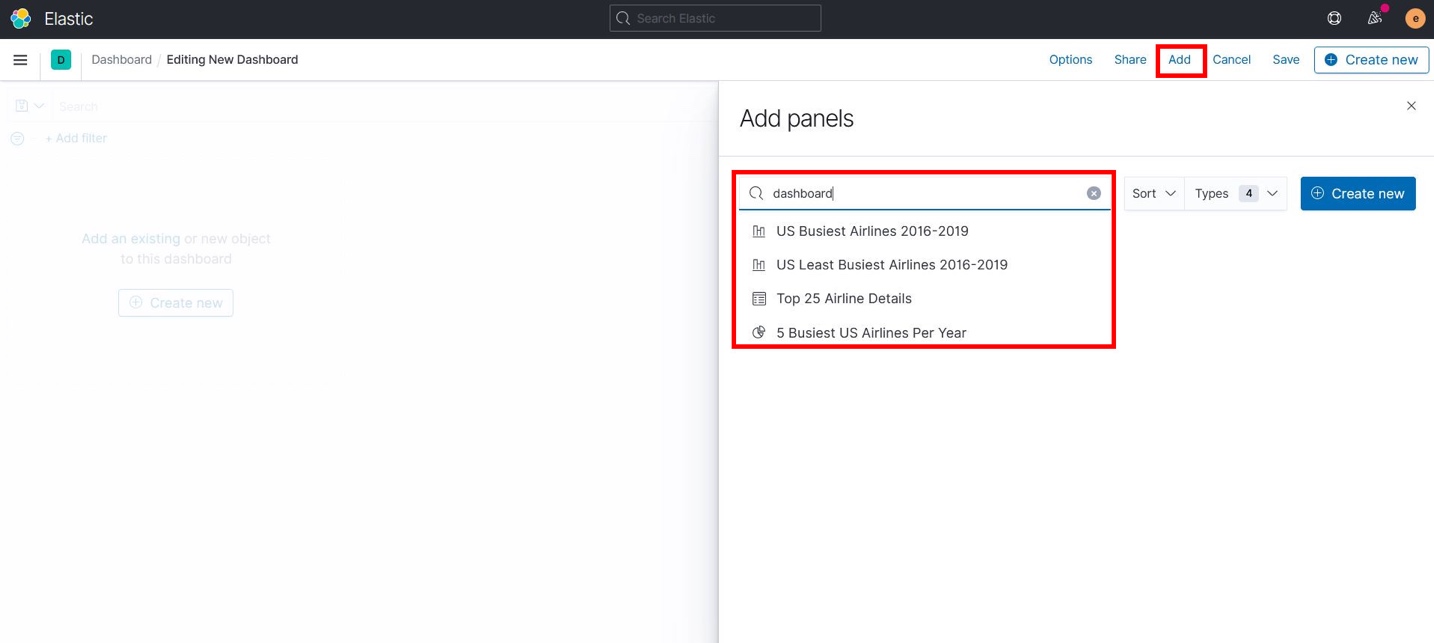
**Dashboard**

You now have all the necessary visualizations to create your Dashboard.

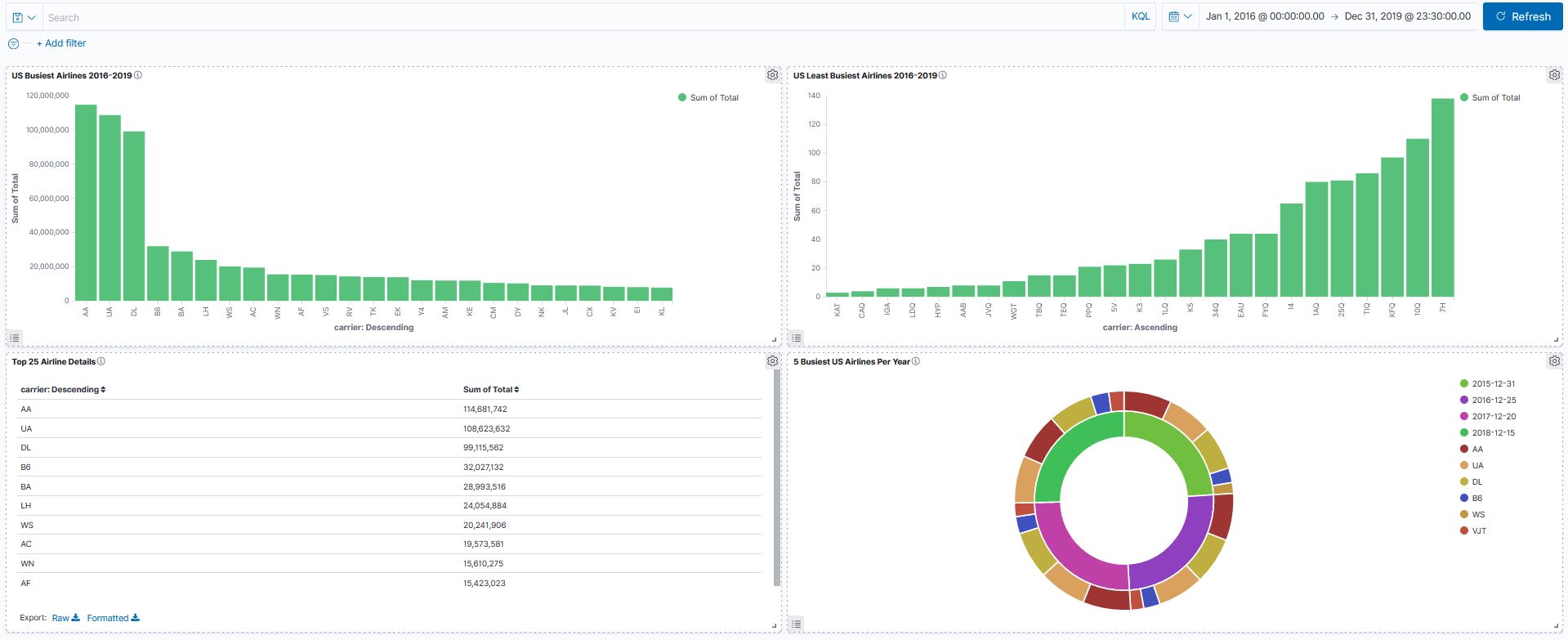
1. On the top left menu go to “Dashboard” under the Kibana section. Click “Create Dashboard.”



1. On the top right menu click “Add” and search for the visualizations you just created, then add them one by one.



1. After adding all four visualizations, your Dashboard should look like this:



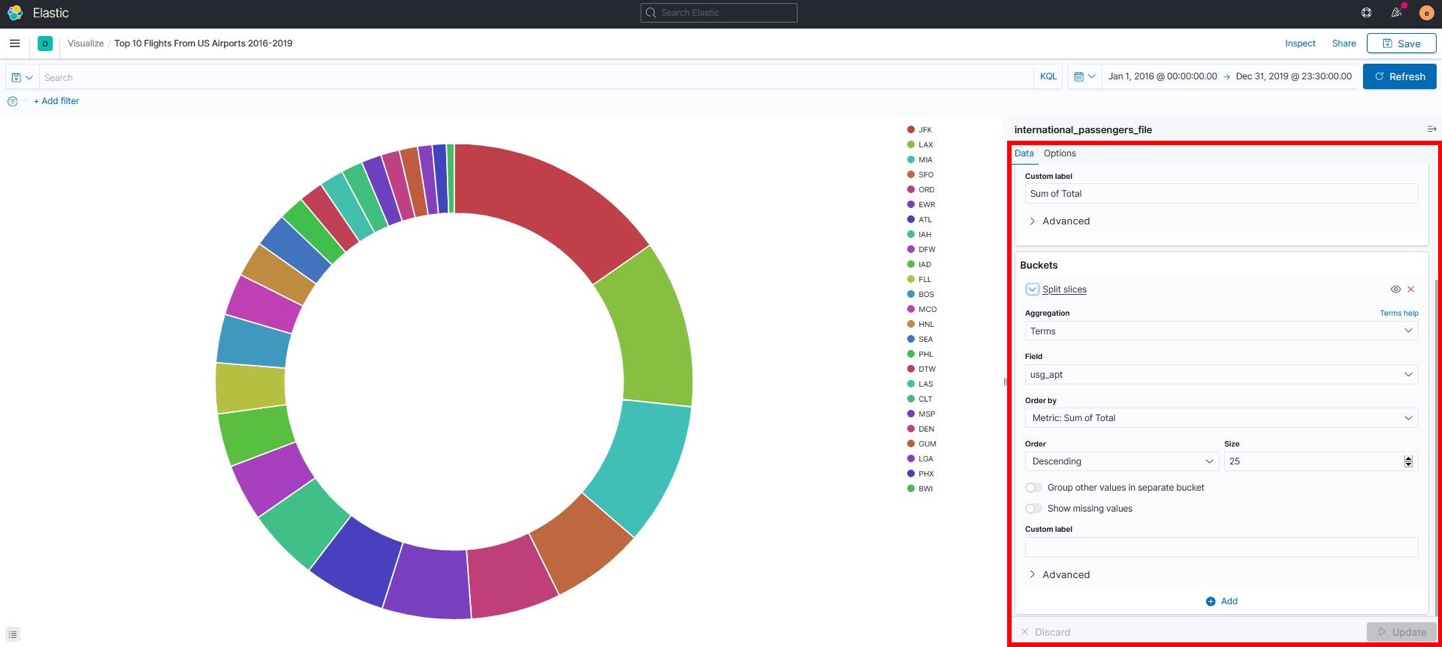
1. Save your Dashboard and add a Title and Description.
2. **Top 10 Flights From US Airports 2016-2016 Pie** Chart
3. This step shows a Pie Chart that demonstrates the Top 10 US Airports With Most Departures from 2016 to 2019, gathered from a list of the top 25 US airports with the most departures.
4. In the “Visualize” section, click “Create Visualization” and select “Pie,” as demonstrated in previous steps. Search for “international\_passengers\_file” and select it as your source.
5. In the “Metrics” pane, use the following settings:

* Aggregation: Sum
* Field: Total
* Custom Label: Sum of Total

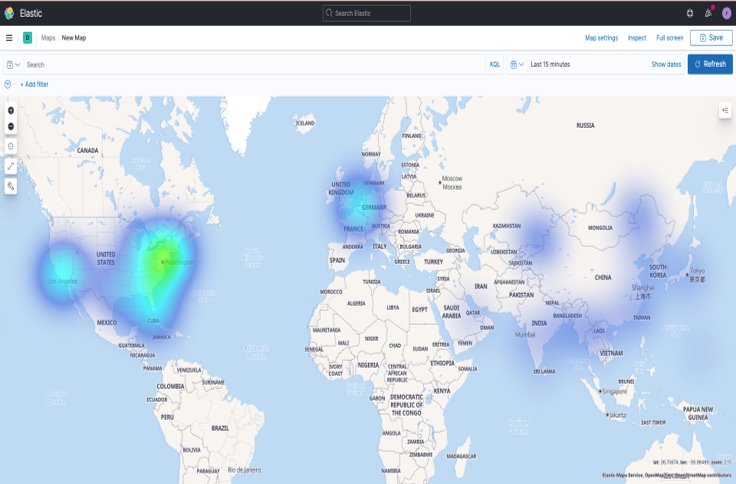
1. In the “Buckets” pane, use the following settings:

* Aggregation: Terms
* Field: usg\_apt
* Order by: Metric: Sum of Total
* Order: Descending
* Size: 25

1. Click “Update,” then save your chart and add a Title and Description. Your chart will look as follows:



1. Heatmap for busiest Airports in the World – for this visualization we extracted the top 25 airports with most passengers from 2016-2019 and the same for the top 25 foreign airports receiving most passengers. We then manually looked up each latitude and longitude coordinate and created a sub (smaller dataset) that would be used as the base for our map below.
2. In the visualize section we selected maps.
3. Next, we fixed the time range, so data displays from January 1, 2016-December 31, 2019.
4. Next, we selected add layer.
5. We chose Heatmap as our option.
6. Lastly, we chose the index pattern, which is our uploaded dataset.
7. Your chart will look as follows:



1. This step shows a Vertical Bar graph with the Top 10 Foreign Airports with The Most Arrivals From 2016 to 2019, gathered from a list of the top 25 foreign airports with the most arrivals.

**Top 10 Flights To Foreign Airports 2016-2019 Vertical Bar Graph**

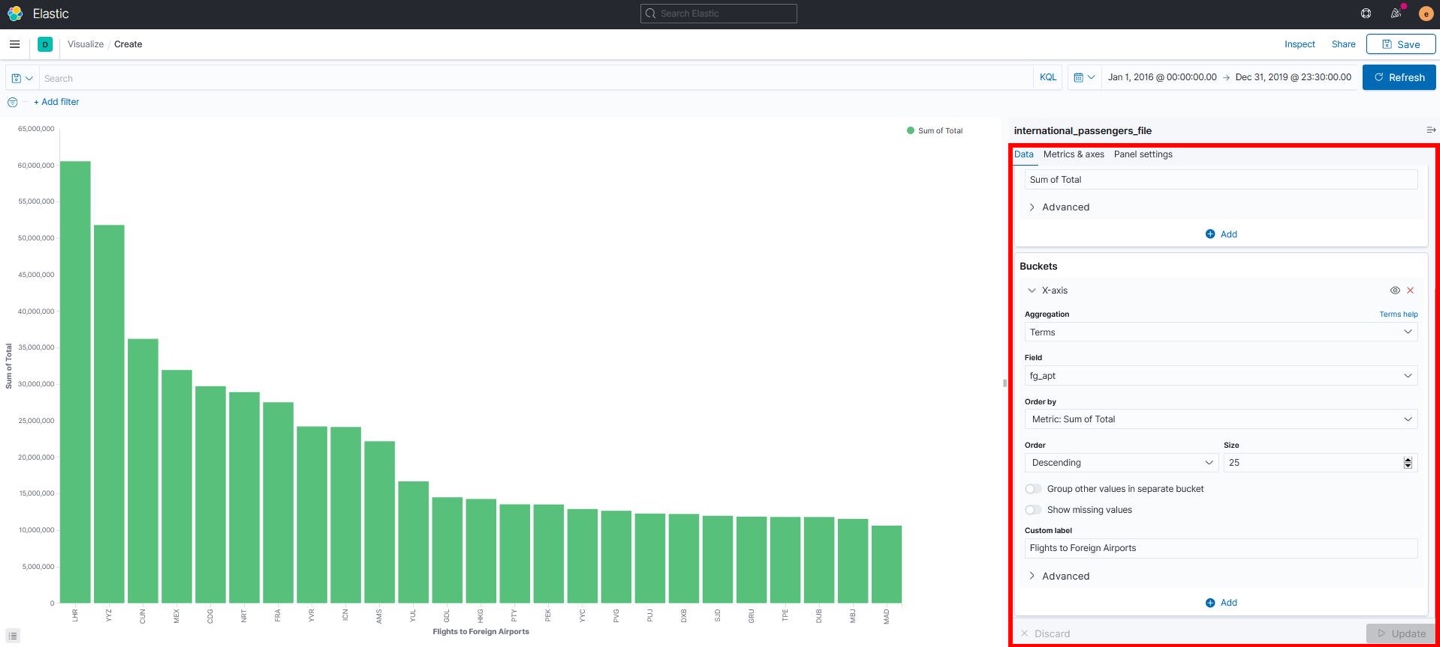
1. In the “Visualize” section click “Create Visualization” and select “Vertical Graph.”
2. Use “internatinal\_passengers\_file” as the source.
3. In the “Metrics” pane, use the following settings for the Y-axis:

* Aggregation: Sum
* Field: Total
* Custom Label: Sum of Total

1. In the “Buckets” pane, use the following setting for the X-axis:

* Aggregation: Terms
* Field: fg\_apt
* Order by: Metric: Sum of Total
* Order: Descending
* Size: 25

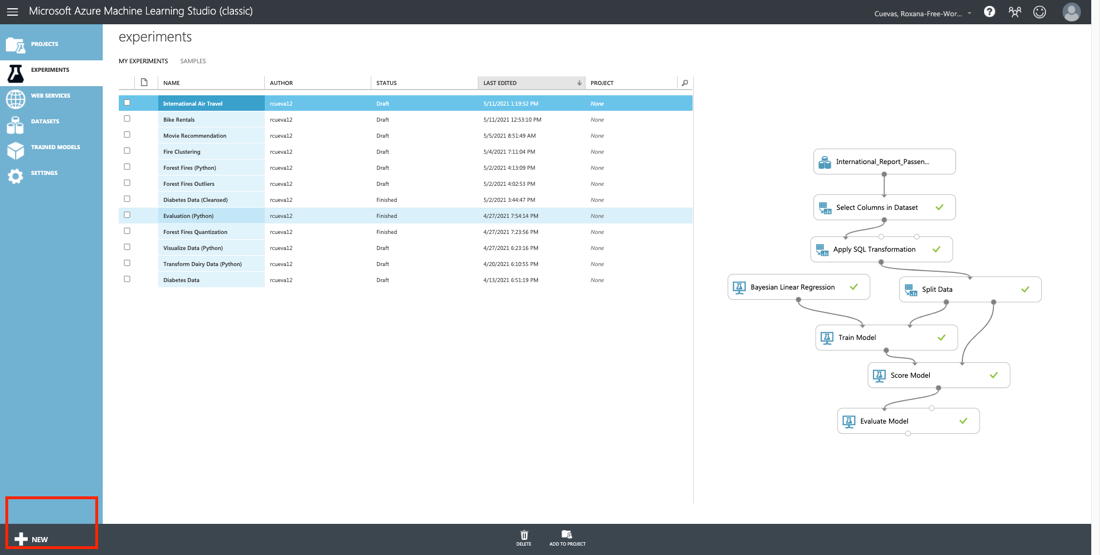
1. Click “Update,” then save your graph and add a Title and Description. Your graph will look as follows:



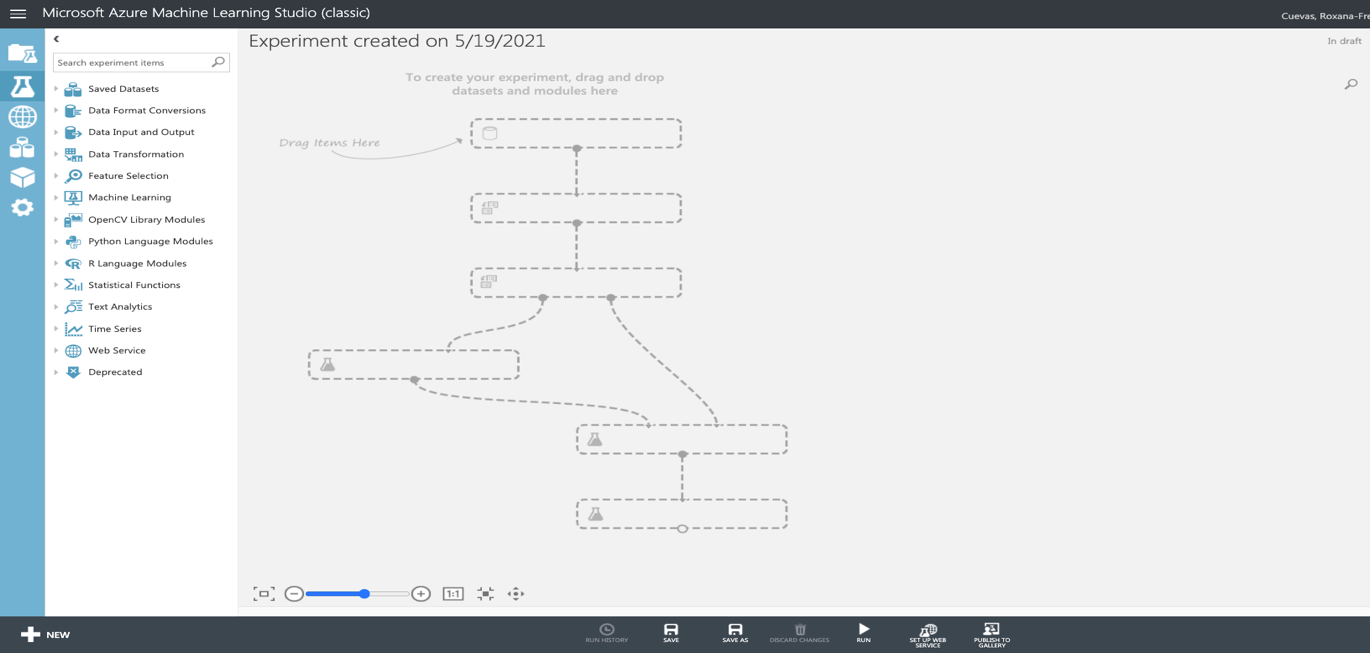
Step 3: Azure Predictive Analysis

This step is to load a dataset into Azure then build and train a predictive model for this data.

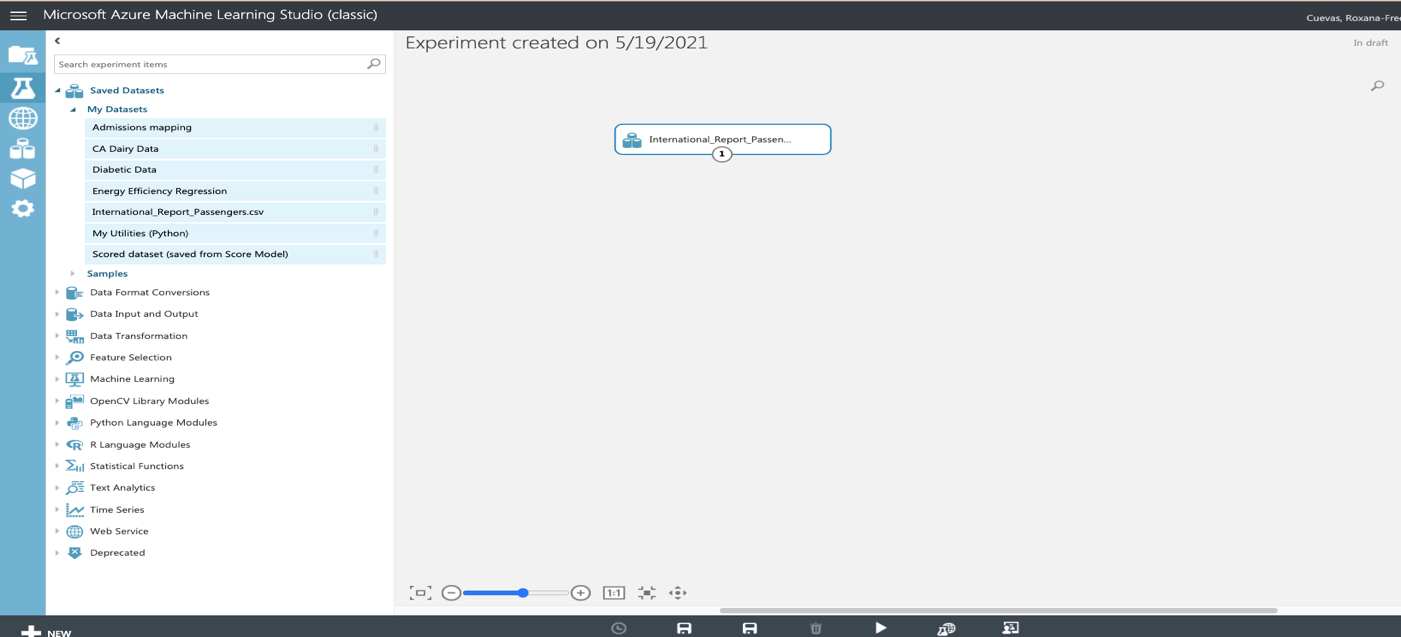
1. Open a new browser and login to your account for <https://studio.azureml.net/>
2. First we will upload our dataset csv file into Azure. In the Studio, at the bottom left click **NEW**.



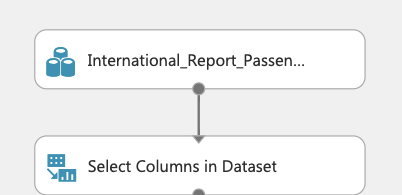
1. Then select **Dataset**
2. Then select **from Local File**.
3. Next click choose file and locate your dataset from your computer folders. Once dataset completes uploading it will be available to use for future projects.
4. Now, In Studio on the left bottom, click **NEW** and select **Blank Experiment**. This creates a blank experiment which looks similar to the following image.



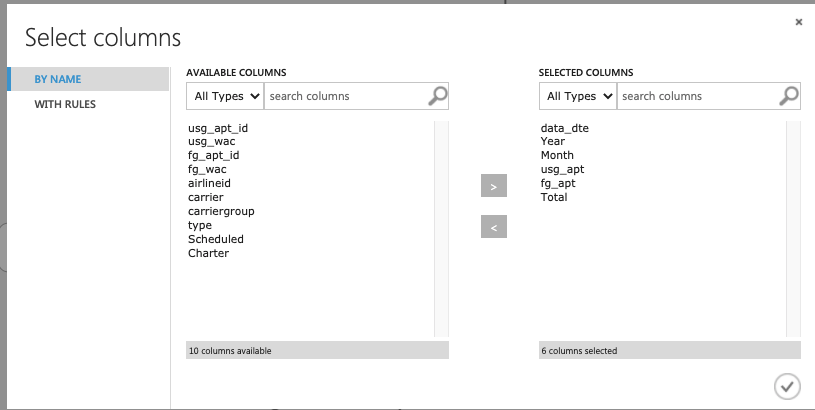
1. Change the title of your experiment to International Air Travel
2. In the experiment's items pane on the left, expand **Saved Datasets**, expand **My Datasets**, and drag I**nternational\_Report\_Passenger.csv** to the experiment canvas in the middle of the page as shown in the following image.



1. Select I**nternational\_Report\_Passenger on the canvas** and note that it has a single output port (indicated as a circle containing the value **1** at the bottom of the dataset icon). Right-click this output port and click **Visualize** to see the data that the dataset contains.
2. Close the dataset, and in the experiment items pane, in the search box, type **Select Columns in Dataset**. Then, in the filtered experiment items pane, under **Data Transformation** and **Manipulation**, drag the **Select Columns in Dataset** module to the canvas and place it under the I**nternational\_Report\_Passenger.**
3. Click the output port of the I**nternational\_Report\_Passenger,** and drag it to the input port at the top of the S**elect Columns in Dataset** module to connect the items. Your experiment should now look like the following image.



1. Select the **Select Columns in Dataset** module, and in the **Properties** pane on the right, click **Launch column selector**. We will now choose the columns we want to have passed (or *projected*) into the data flow for the next module. From available columns, choose *date\_dte*, *Year*, *Month*, *usg\_apt*, *fg\_apt*, and *Total* as shown below.



1. On the toolbar at the bottom of the page, click **SAVE** to save the experiment. Then click **RUN** to run the experiment.
2. When the experiment has finished running, note the status displayed at the top-right of the experiment canvas and the green checkmark that indicates that the **Select Columns in Dataset** module completed successfully.
3. In the experiment items pane, search for “**Apply SQL Transformation**”. Then drag the **Apply SQL Transformation** to the experiment canvas, under the **Select Columns in Dataset**, and connect the **Select Columns in Dataset** to the leftmost input port of the **Apply SQL Transformation**.
4. Select the **Apply SQL Transformation** module, and in the **Properties** pane, replace the default SQL script with the following code.

SELECT \*, Total\*Total AS Totsqrd FROM t1

WHERE Month BETWEEN 1 AND 12;

This code returns all columns for the rows passed to the input port in which the **Month** column value is between 1 and 12. This removes any rows that contain an invalid month value. Additionally, it generates a new column named **Totsqrd** that contains the total passengers value squared.

1. Save and run the experiment. At each node, the green circling should be shown while it is running. Then, when the experiment has finished with “Finished running” at the top of the pane, visualize the **Results dataset** output of the **Apply SQL Transformation** module, and view the filtered data, as shown here.

Graphical user interface, application

Description automatically generated

1. In the experiment items pane, search for “**Split** **Data**”. Then drag the **Split** **Data** to the experiment canvas, under the **Apply SQL Transformation**, and connect the **Apply SQL Transformation** to the **Split** **Data** as shown below.

Diagram

Description automatically generated

Now we will add and Train a Model

1. In the experiment items pane, search for “**Bayesian Linear Regression**”. Then drag the **Bayesian Linear Regression** module to the experiment canvas, under the **Apply SQL Transformation** next to **Split Data** module.

A picture containing text

Description automatically generated

1. In the experiment items pane, search for “**Train Model**”. Then drag **Train Model** to the experiment canvas, under the **Bayesian Linear Regression** module, and connect the output of **Bayesian Linear Regression** to the leftmost input of the **Train Model**, and the leftmost output of the **Split Data** to the rightmost input of the **Train Model** as shown below.

Text

Description automatically generated

1. Select **Train Model** and **Launch column selector,** to choose which columns will be considered in the prediction calculation. For this dataset we will use the total number of passengers. Choose only the *Total* column as shown below.

Graphical user interface, text, application, email

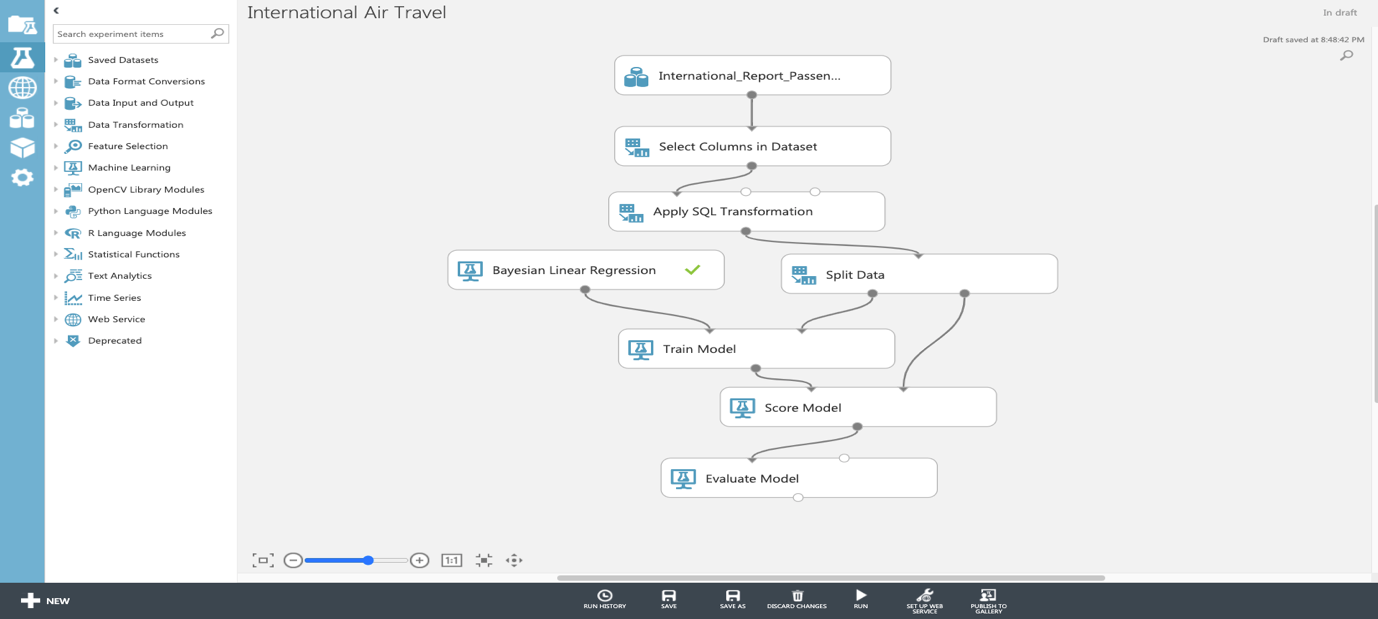
Description automatically generated

1. In the experiment items pane, search for “**Score Model**”. Then drag **Score Model** to the experiment canvas, under **Train Model**, and connect the output of the **Train Model** to the leftmost input of the **Score Model** and the rightmost output of the **Split Data** to the rightmost input of the **Score Model** as shown below.

A picture containing text

Description automatically generated

1. In the experiment items pane, search for “**Evaluate Model**”. Then drag **Evaluate Model** to the experiment canvas, under the **Score Model**, and connect the output of the **Score Model** to the input of the **Evaluate Model** module as shown below. Your final experiment should look as follows. Save and Run the experiment.



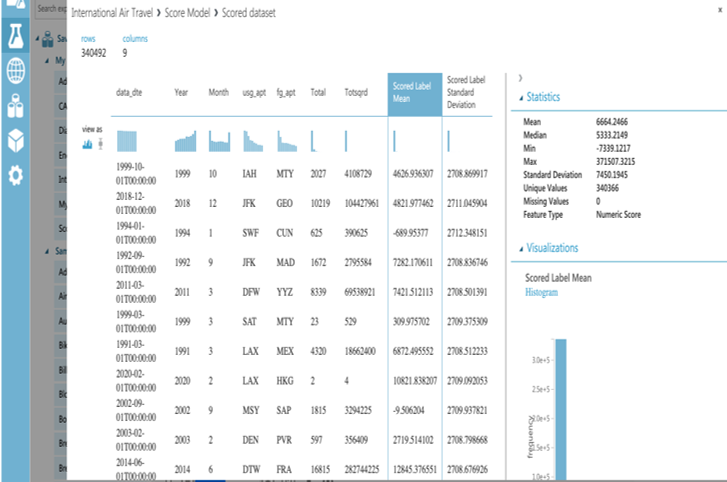
1. When your experiment has finished running, visualize the Evaluate Model Results dataset output.

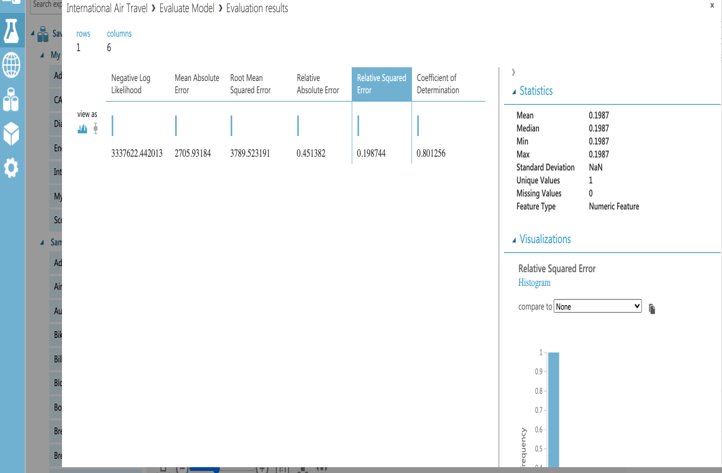
**Note the following:**

Coefficient of Determination = 0.80 **for this value|closer to 1 is better**

Relative Squared Error = 0.19 **for this value|closer to 0 is better**

These values indicate how efficient the model will be in determining the predicted value.





References

* 1. Data Source URL: [https://www.kaggle.com/parulpandey/us-international-air-traffic-data](https://www.kaggle.com/parulpandey/us-international-air-traffic-data%E2%80%8B)
  2. GitHub URL: [https://github.com/rcueva12/CIS3200-Data\_Processing\_Analytics/tree/main](https://github.com/rcueva12/CIS3200-Data_Processing_Analytics/tree/main%E2%80%8B)
  3. Small Dataset for Maps URL: <https://github.com/rcueva12/CIS3200-Data_Processing_Analytics/blob/main/airports4.csv>
  4. Azure URL: <https://studio.azureml.net/Home/ViewWorkspaceCached/17993356d8fa48389e9a5796ceb6afcb#Workspaces/Experiments/Experiment/17993356d8fa48389e9a5796ceb6afcb.f-id.1bc20085248a4dba86f25479eb229049/ViewExperiment>
  5. Kibana URL: <https://e4370728e6b04da7a2f0204627978298.us-central1.gcp.cloud.es.io:9243/app/home#/>
  6. Skyscanner: <https://www.skyscanner.com/tips-and-inspiration/best-time-to-book-usa>
  7. Wall Street Journal: <https://www.wsj.com/articles/the-best-and-worst-u-s-airlines-of-2020-11611756016>