

Churn Modeling Charts

Last amended: 25th Nov, 2025

Reference: [Chart types in Atlas](#)

My folder: D:\OneDrive\Documents\mongodb

GitHub repo: [Databases](#)

Churn modeling data [is here](#) in github

Table of Contents

1.	Saved Dashboard	3
2.	YouTube Videos:	3
3.	Our Dataset:.....	3
4.	Our DatasetData type changes:	3
5.	Bin Data.....	4
6.	Create Grouped Column/Bar charts (Ref link here).....	4
	What is a Series?.....	4
	Create Grouped Column chart.....	5
	Create Grouped Column chart with Series	6
	Plotting distribution of a numeric feature	6
7.	Stacked Column chart with Series	7
8.	Create Grouped Bar chart with two aggregations	8
9.	Heatmap: Two categories + one number	8
10.	Geospatial map (See this Link).....	9
	Choropleth map	9
	Heat map geospatial charts	10
	Scatter plot geospatial charts.	10
11.	Scatter plot (see Link here)	12
	All countries selected.....	12
	Only one country selected	13
	Applying Filters.....	13
12.	Top-item chart (see Link here).....	14
	Word Cloud (See link here)	17
13.	Discrete Area Chart (see link here)	19
	Binned-age charts with Binned-balance as a <i>Series</i>	19
	Binned-Age charts with geography as Series.....	20
14.	How to change legend colors?.....	21
15.	Import/export a Dashboard	22
16.	Refresh for streaming data	23

Dataset used is the file: `churn_modeling.csv`. It is available in the VM under folder: `/home/ashok/Documents/mongodb/datasets/`. Churn Modeling dataset possibly contains **duplicate** records. For example, `customerid: 15616555` occurs twice. Filter using: `{"customerId" : 15616555}`, For each customerId, there should be just one row.

Two columns, latitude and longitude, were not part of this dataset originally. However, they have been created by adding some randomization to latitude/longitude of each one of the three countries.

1. Saved Dashboard

The dashboard built here is available in the VM and in My folder (on OneDrive), as: **churnModeling.charts**.

2. YouTube Videos:

See [this YouTube video](#) to help you in understanding how cg=harts are cerated in Atlas. Also see [this video](#).

3. Our Dataset:

This is an extract of our dataset. We have imported it into Atlas through Compass.

Row Number	Customer Id	surname	Credit Score	geography	gender	age	tenure	balance
1	15634602	Hargrave	619	France	Female	42	2	0
2	15647311	Hill	608	Spain	Female	41	1	83808
3	15619304	Onio	502	France	Female	42	8	159661
4	15701354	Boni	699	France	Female	39	1	0
5	15737888	Mitchell	850	Spain	Female	43	2	125511
6	15574012	Chu	645	Spain	Male	44	8	113756

num Of Products	has Cr Card	isActive Member	estimatedSalary	exited	Latitude	Longitude
1	1	1	101349	1	46.48	4.40
1	0	1	112543	0	41.82	-2.37
3	1	0	113932	1	48.32	3.98
2	0	0	93827	0	47.81	4.12
1	1	1	79084	0	41.82	-2.04
2	1	0	149757	1	41.39	-3.61

4. Our DatasetData type changes:

. At times, datatypes of imported data need to be changed to draw some charts. Change types of `numOfProducts`, `customerId`, `hasCrCard`, `isActiveMember`, `tenure` and `exited` only as and when required in the specific chart. All the six fields are numeric so these can be easily changed to `String` data types. Our all conversions of data types in this example dataset are from integer (or `numeric`) to `String`.

5. Bin Data

(See [here](#) for various types of binning.). Atlas Charts supports binning date, numeric, and string fields in your data. Binning breaks continuous data into discrete groups called *bins*, with each bin containing a contiguous subset of the original values. For example, age of users ranging from 15 to 80 may be binned as 15 to 30, 30 to 50 and 50 to 80. Binning, thus creates three categories of age: young, middle and senior.

6. Create Grouped Column/Bar charts ([Ref link here](#))

Both charts are same except that orientations of bars are different. Briefly, group data by a category and draw bars with length proportion to some aggregated data value. Column and bar charts provide high-level overviews of data trends by comparing values within a specific category.

Encoding Channel	Description
X / Y Axis	<ul style="list-style-type: none">In the category axis encoding channel, a bar or column is rendered for each unique value from this field. <p>In a column chart, the category is the X Axis, which results in vertical segments. In a bar chart, the category is the Y Axis, which results in horizontal segments.</p> <ul style="list-style-type: none">The aggregation axis encoding channel dictates which field to aggregate on and the type of aggregation to perform. This ultimately dictates the size of each bar or column. <p>You can aggregate upon multiple fields in your dataset to create multi-series charts. For more information, see Multiple Field Mappings.</p>
Series	<p>(Optional) A category encoding channel that segments the axis category based on the unique values in this field.</p> <p>If omitted, Atlas Charts renders a basic bar or column chart with just a comparison of the X Axis and Y Axis fields.</p> <p>This option is only available for Grouped and Stacked charts with a <i>single</i> field mapped in the aggregation channel.</p>
Color	<p>(Optional) An aggregation encoding channel for Colored charts that changes the bars or columns' colors to reflect the aggregated value of the field, with darker colors meaning greater values.</p>

What is a Series?

A series (optional) further segments *Category* channel into subcategories as per levels or values in Series. Series creates subgroups within each group. Example:

Let Category names be: Income; Levels: A, B, C

Let Aggregation Channel be: Age. So grouped graph (bar or column) will display mean age at Levels A, B and C. Add now another category in *Series* Channel.

Category Name: Type of car; Levels: XX, YY, ZZ, UU:

Now each one of the columns A, B, C (for grouped columns chart) will get further subdivided, as:

A: A + XX Mean age

A	A + YY	Mean age
A	A + ZZ	Mean age
A	A + UU	Mean age

B:	B + XX	Mean age
B	B + YY	Mean age
B	B + ZZ	Mean age
B	B + UU	Mean age

Create Grouped Column chart

Objective: Group by *hasCrCards* and find out how many have credit cards and how many do not? Further, create sub-groups *tenure* wise within each group of *hasCrCards* and count.

Fields	Whether converted	Where?	Operation
<i>hasCrCards</i>	Yes	X-axis	
<i>_id</i>	No	Y-axis	<i>count()</i>
<i>tenure</i>	Yes	Series	

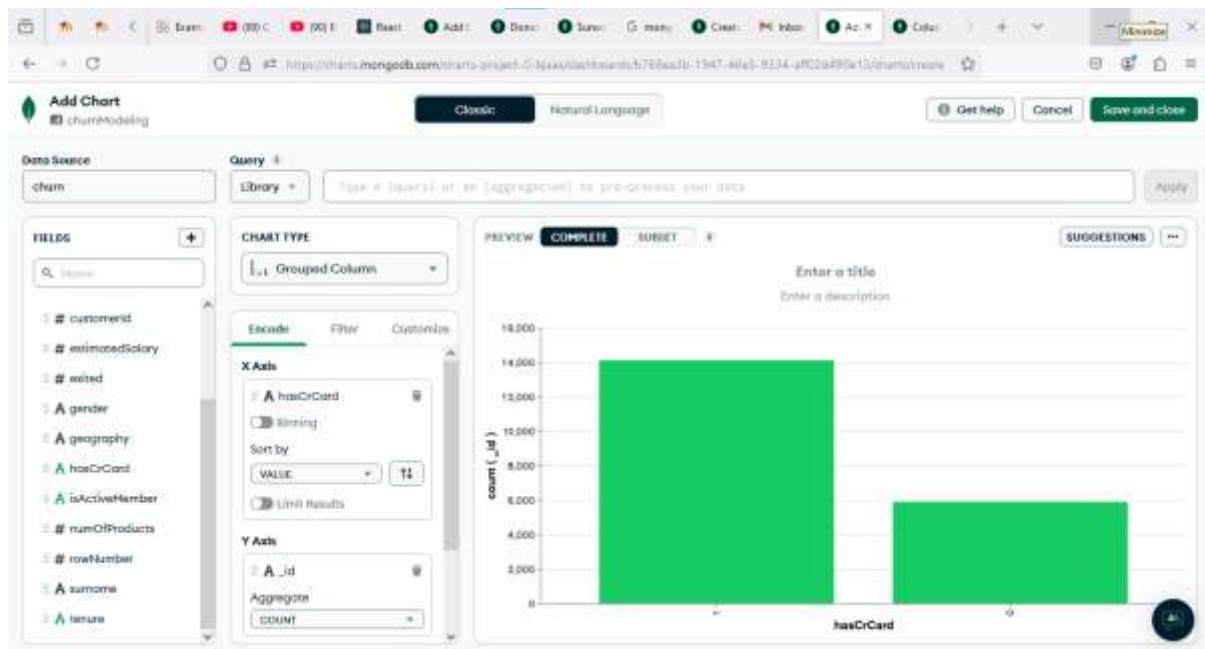


Figure 1 :Group by *hasCrCards* and count hHow many have credit cards and how many do not?

Create Grouped Column chart with Series

Add *tenure* as a *Series* to create subgroups. Each column of *hasCrCard* is further subdivided, *tenure* wise.



Figure 2: How many have credit cards and how many do not, tenure wise?

Plotting distribution of a numeric feature

Atlas charts do not have density plots. Instead one has to build a histogram (not exactly histogram as bars are not horizontally joined) through grouped column chart.

Objective: An example histogram in the case for age is below:

Fields	Whether converted	Where?	Operation
age	No	X-axis	Bin size: 10 (Note: It is bin-size NOT bin count)
age	No	Y-axis	count()

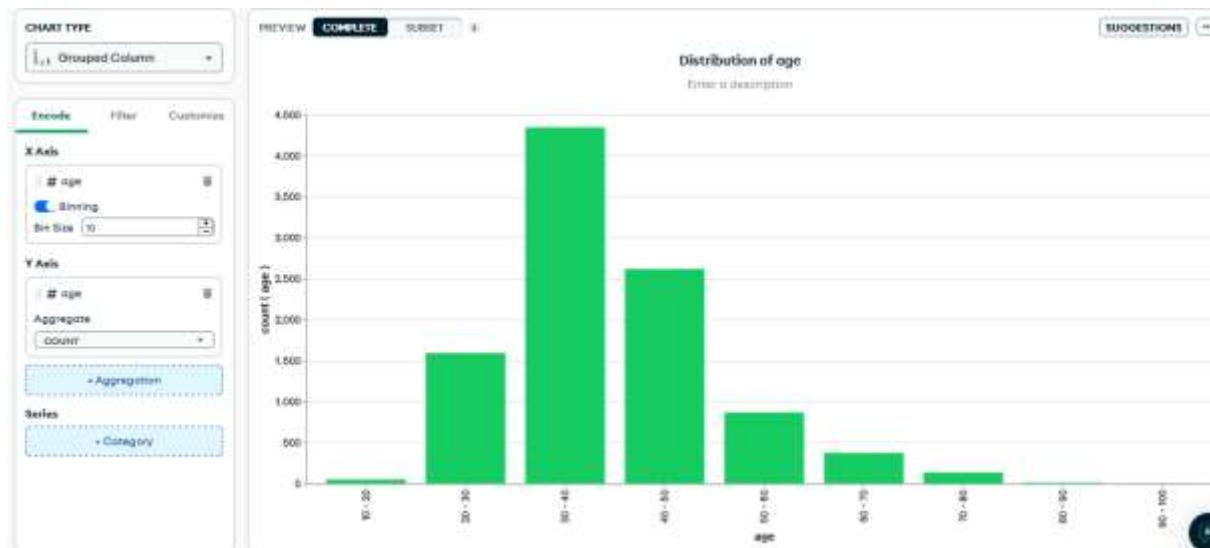


Figure 3: Grouped column chart displaying histogram for age

7. Stacked Column chart with Series

Instead of showing subgroups side-by-side, Stacked column chart shows them within each bar or group. The earlier graph could also be transformed into stacked bar/column chart as follows. Aggregated values of each *Series* levels now get displayed in stacked manner instead of side-by-side bars.

Objective: Group by geography and within each region further sub-group tenure wise to show mean of balance.

Fields	Whether converted	Where?	Operation
Geography	No	X-axis	
balance	No	Y-axis	mean()
tenure	Yes	Series	

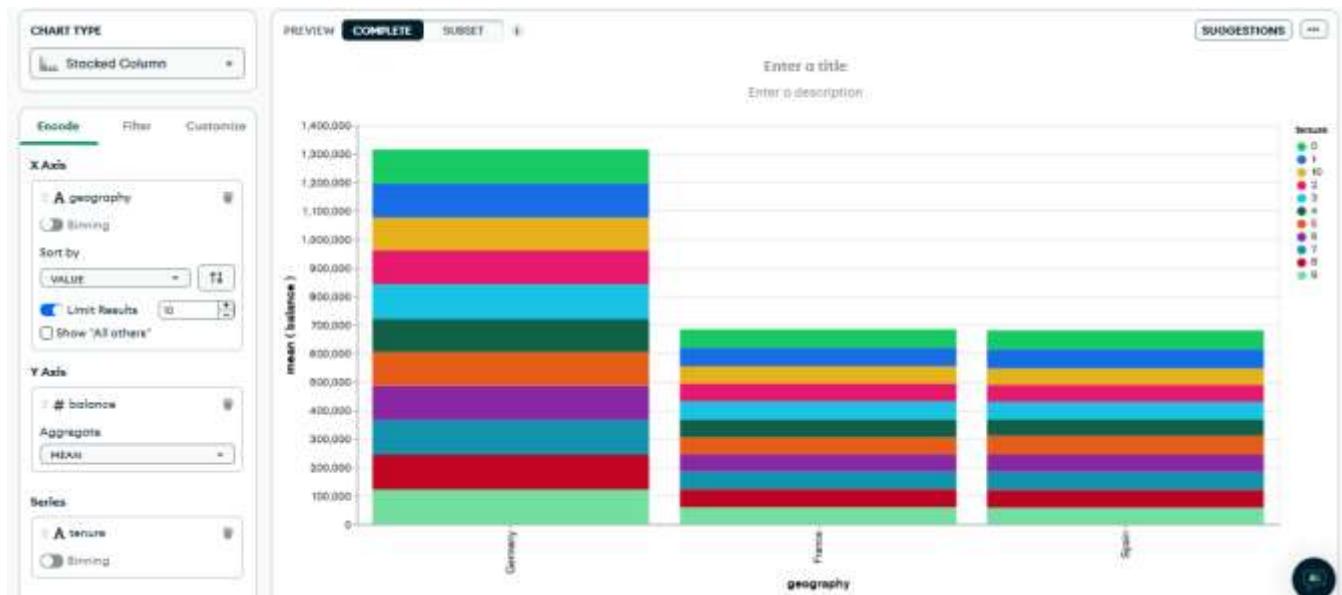


Figure 4: Aggregate balance against Geography and against tenure.

8. Create Grouped Bar chart with two aggregations

Following grouped bar chart shows aggregation of *creditScore* and *age* against top-10 *customerIds*. Carefully observe that on X-axis, we have aggregated on two columns; As there is a doubt regarding duplicate columns, even though *customerId* is unique, we have taken MEAN as aggregate metric and not SUM. Also see Arrow symbols beside *Sort by* value. Under *Sort By*, select *Mean Credit Score*.

Objective: Group by *customerId* and display mean *creditScore* as also mean *age*.

Fields	Whether converted	Where?	Operation
creditScore	No	X-axis	mean()
age	No	X-axis	mean()
customerId	Yes	Y-axis	Limit Results: 10 Play with Arrows besides Sort By



Figure 5; Two aggregations on X-axis. Limit customerId to 10. Play with Arrows beside the *Sort By* select Mean Credit Score. Instead of Value.

9. Heatmap: Two categories + one number

Objective: Show mean *balance*, *exited* and *gender* wise.

Fields	Whether converted	Where?	Operation
exited	Yes	X-axis	
gender	No	Y-axis	
balance	No	Intensity by	mean()



Figure 6: Heatmap between two categories and mean balance

10. Geospatial map (See this Link)

Choropleth map

Objective: From which region of the World there have been most and least exits.

Fields	Whether converted	Where?	Operation
geography	No	location	
exited	No	color	sum()

Change color palette to Diverging under Customize

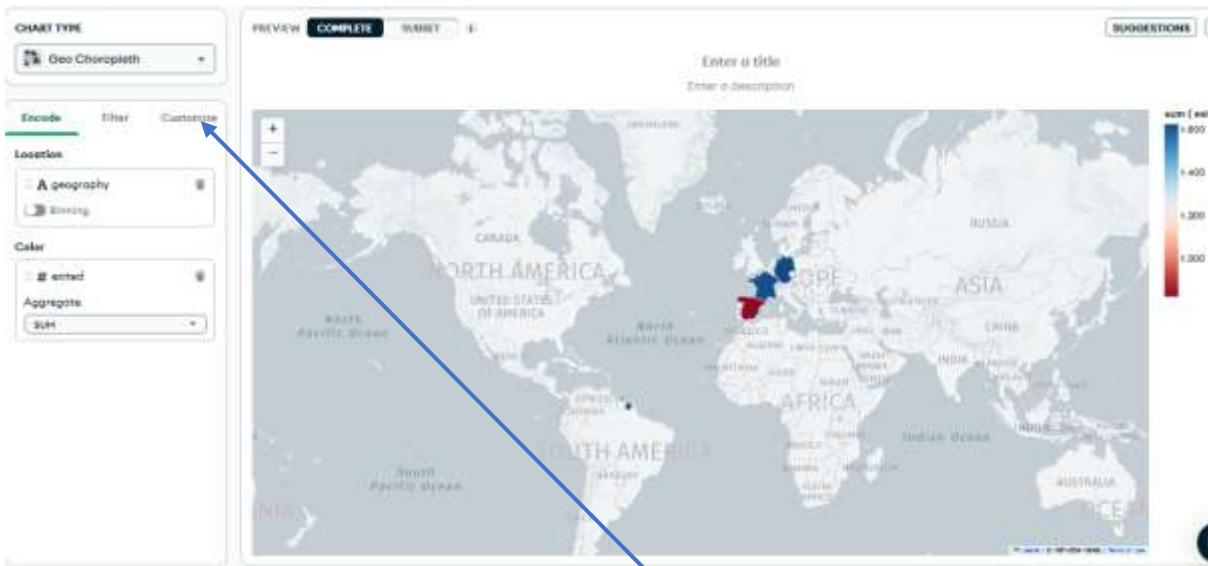


Figure 7: Customise colour palette to make it diverging under Customize

The following figures are examples of other geospatial maps **IF** map-coordinates (lat/long) are available.

Heat map geospatial charts

(see more [details here](#))

One can use geospatial heatmap charts to visualize how geographic areas of a map compare to one another based on a given criterion. Intensity is an aggregation metric that changes the shading of the colors in different chart regions to reflect the aggregated values of this field. More intense colors in the visualization reflect greater aggregated values.

Objective: Show the distribution of variance of *estimatedSalary* in different parts of the World.

Fields	Whether converted	Where?	Operation
Latitude	No	coordinates	
longitude	No	coordinates	
estimatedSalary	No	Intensity	variance()



Figure 8: Heatmap showing *estimatedSalary* by intensity of color.

Scatter plot geospatial charts.

Fields	Whether converted	Where?	Operation
Latitude	No	coordinates	
longitude	No	coordinates	
isActiveMember	No	Size	Mean()
hasCrCard	Yes	Color	
Change color palette for hasCrCard			



Figure 9: This plot is NOT from churn modeling data. Just an example to show possibility.

An expanded region where you can clearly see data-points and which places have more red colored points.



Figure 10: Certain areas have more red colored data points than others

11. Scatter plot (see Link here)

Objective: Relationship between *creditScore* and *balance*. Color each data-point by *geography*.

Fields	Whether converted	Where?	Operation
creditScore	No	X-axis	
balance	No	Y-axis	
geography	No	Color by	
Select subset tab to avoid crowding			

All countries selected

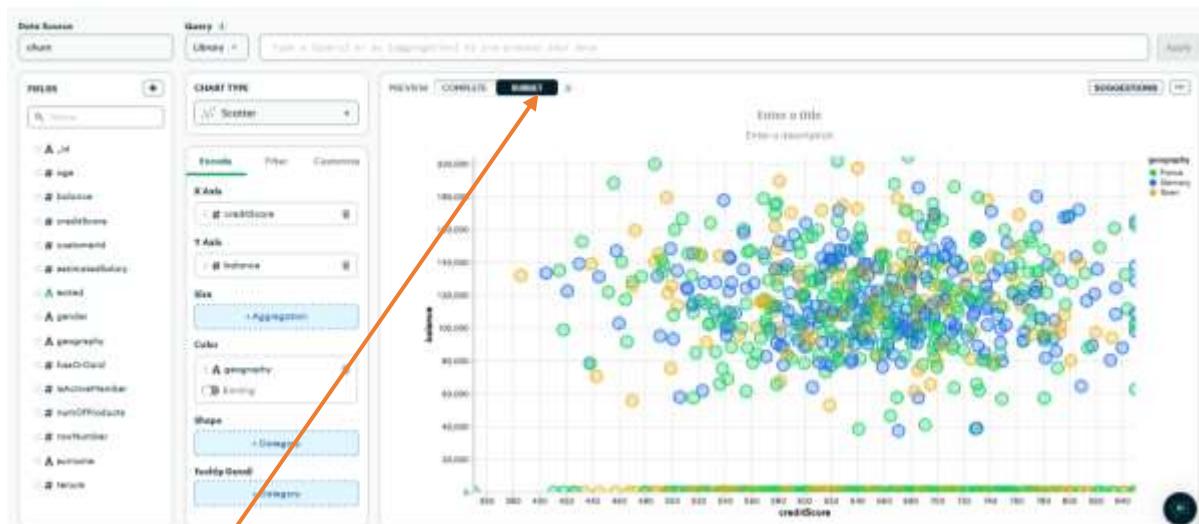


Figure 11: Click on Subset tab first

Only one country selected

For all countries, the trend is as *creditScore* increases, *balance* decreases. For, example, selecting Germany in the legend, shows the following. Under Customization tab, a trend line can also be drawn.

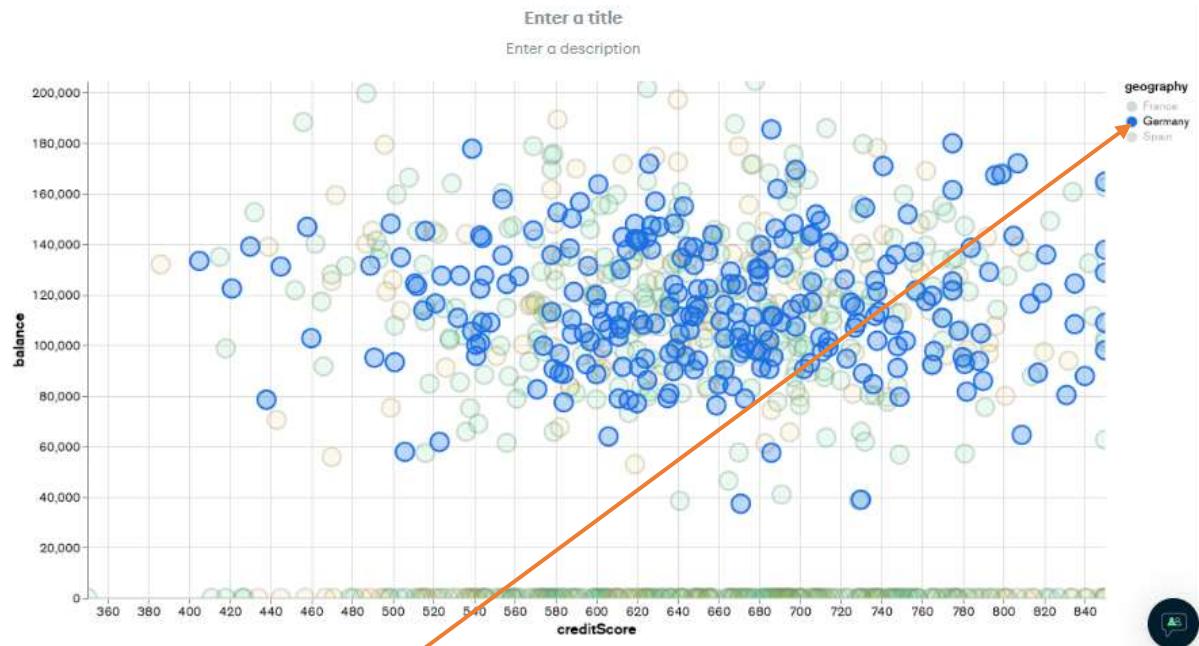


Figure 12: A downwards trend when Germany is selected.in geography legend.

Applying Filters

You can customize the above chart and clearly see a trend line by applying filter(s). For example, in the following chart two filters have been applied. One on *geography*, selecting only *Germany* and the other on *balance* (*balance* > 20000). Next, under *Customise*→*General*, select *Trend line* (see the next figure).

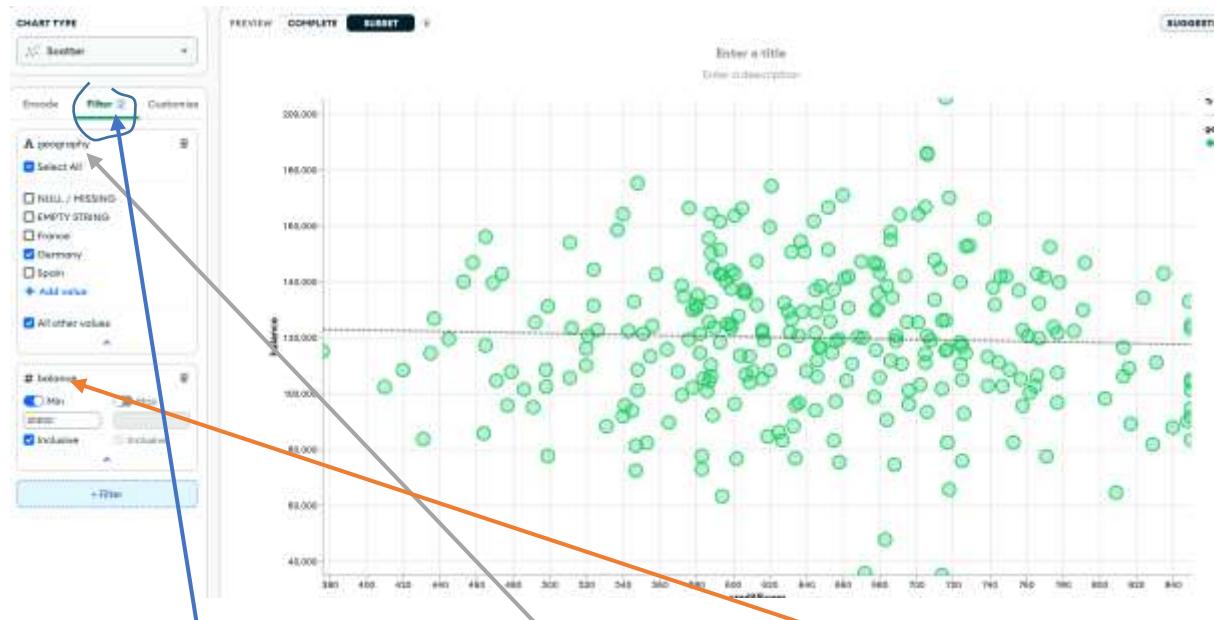


Figure 13: Two filters have been applied. One on geography and the other on balance.



Figure 14: To draw a trend line, click Customise tab. The under General, switch on the trend line.

12. Top-item chart (see [Link here](#))

The following chart first sorts the documents based on `monthly_price` in descending order, and then returns the FIRST DOCUMENT appearing in the sort.

This chart displays the following fields from the SAME first document (i.e. having the highest `monthly_price`):

- `monthly_price`
- `property_type`
- `country`
- `suburb`



Use Top Item charts to show information about the most extreme or most recent instance of a particular field value. For example, you can use Top Item charts to show information on:

- The most recent reading from a collection of sensor data, such as temperature readings.
- The highest rated movie in a collection.
- The cheapest available room in a collection of available hotel listings.
- The largest sale recorded in the past month (when combining a Top Item chart with [date filters](#)).

The following figure is from the churn modelling data. Click the arrows besides *Sort* to see the least case of *estimatedSalary*.

Objective: From the (first) document having maximum (or least) *estimatedSalary* display information of other features.

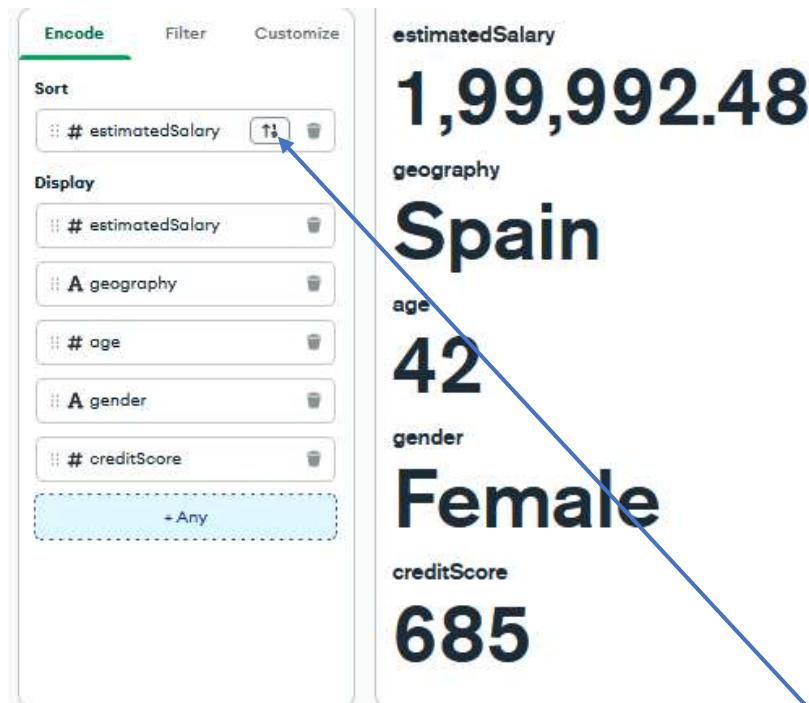


Figure 15: From the Document having highest estimated salary, display desired information. Click here on the arrows to see data on lowest estimated salary.

Word Cloud (See link here)

Fields	Whether converted	Where?	Operation
geography	No	Text	
balance	No	Size	mean()
geography	No	Color by	

Word clouds visually represent text data, highlighting prevalent keywords and phrases. Size of a word dictates relative importance of aggregated field relative to others. The results of the aggregation define the size of each Text value, with larger aggregated values resulting in larger text sizes.

Objective: For each one of the three countries, write their names with font sizes in proportion to mean(balance).

In the following word cloud, Germany has the highest mean(balance) followed by Spain and France.



Figure 16: Germany has the highest mean(balance) followed by Spain and France

Incidentally, the same information can also be seen in the following grouped column chart.

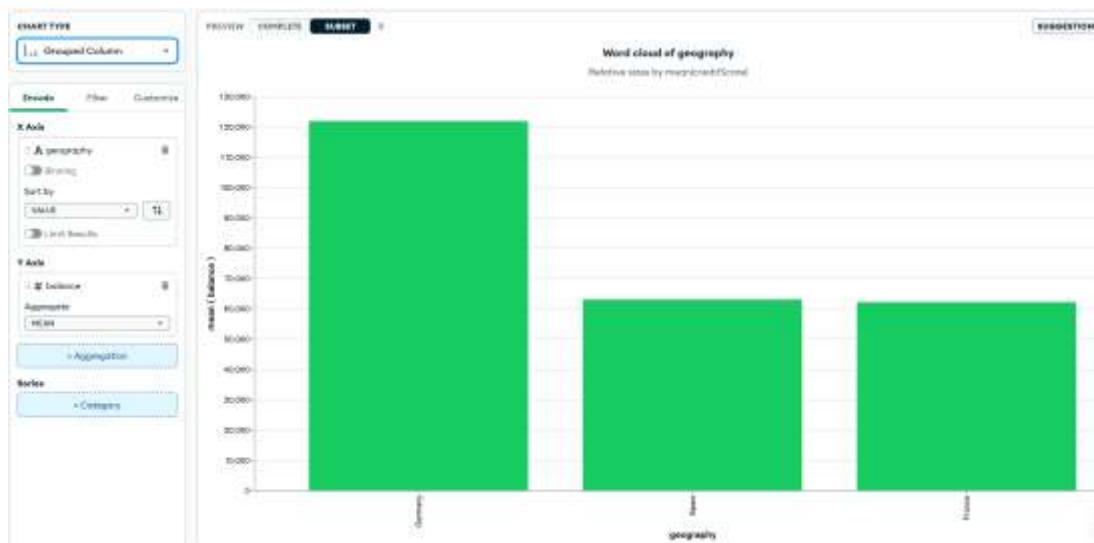


Figure 17: Germany has the highest mean(balance) followed by Spain and France.

13. Discrete Area Chart (see [link here](#))

Binned-age charts with Binned-balance as a *Series*.

When a numeric field is dropped where Atlas expects a *Category*, the numeric field is automatically binned. It is up to us to decide bin-size (but NOT number of bins).

Fields	Whether converted	Where?	Operation
age	No	X-axis	Bin size: 10
age	No	Y-axis	count()
balance	No	Series	Bin size: 40000

Discrete area charts are *stacked line charts*, meaning Atlas Charts plots each series above or below the others in the visualization. The chart shows the total aggregated value of all of the series, so you can easily see the *proportion* of each series **in relation** to the total.

Objective: Distribution of age, balance-wise. For different age-groups show count. Additionally, through color, show counts for each binned group of balance.

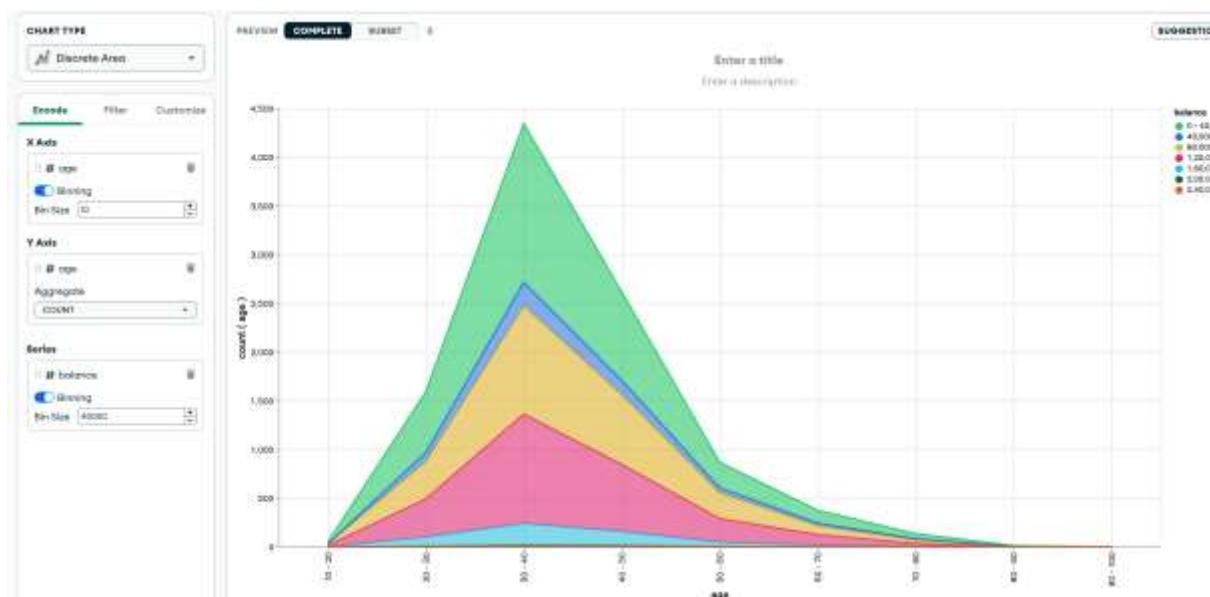
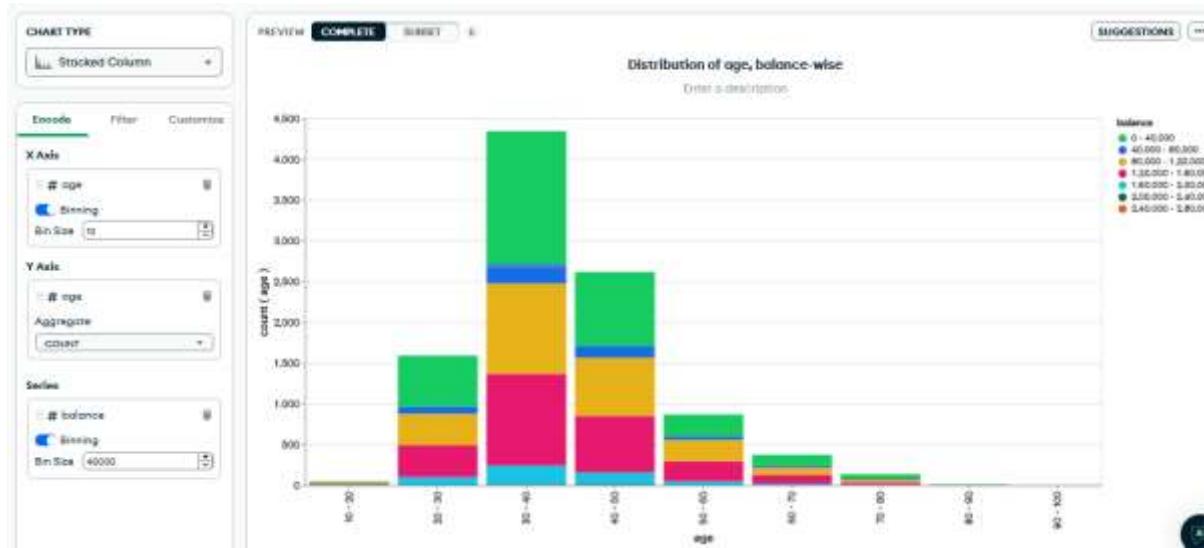


Figure 18: Area charts are stacked. Thus, blue colored category shows the least count among all age groups.

The following graph shows the same information (and through the same color palette) as above but through stacked column chart.:



Binned-Age charts with geography as Series.

Another area chart is below. It is *geography* wise. France has the highest overall proportions among all the age groups with Spain and Germany almost equal.

Objective: Distribution of age, geography wise

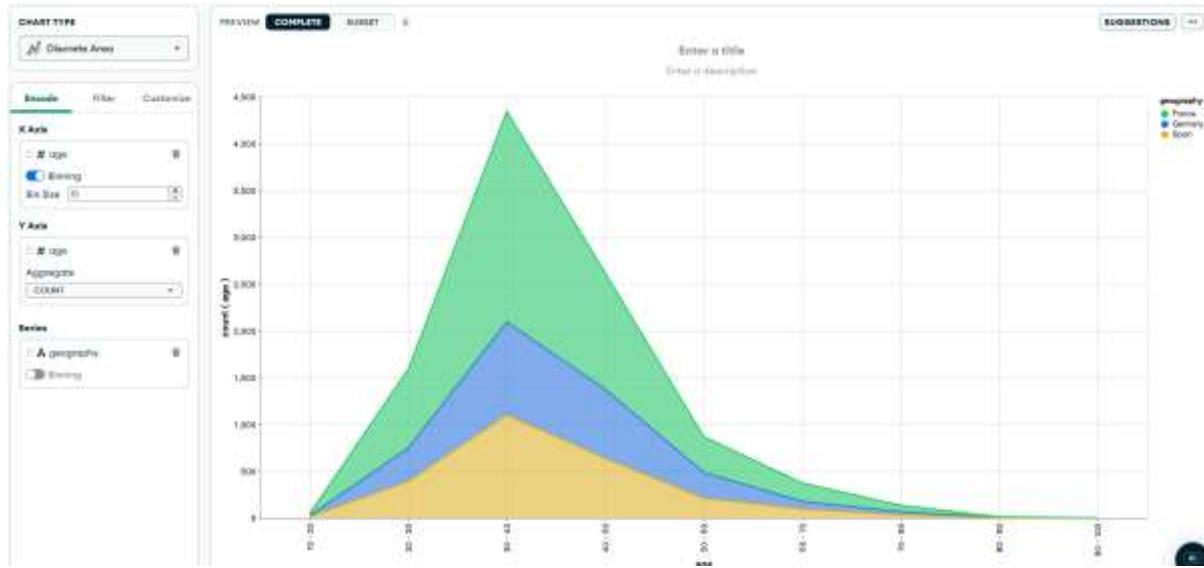
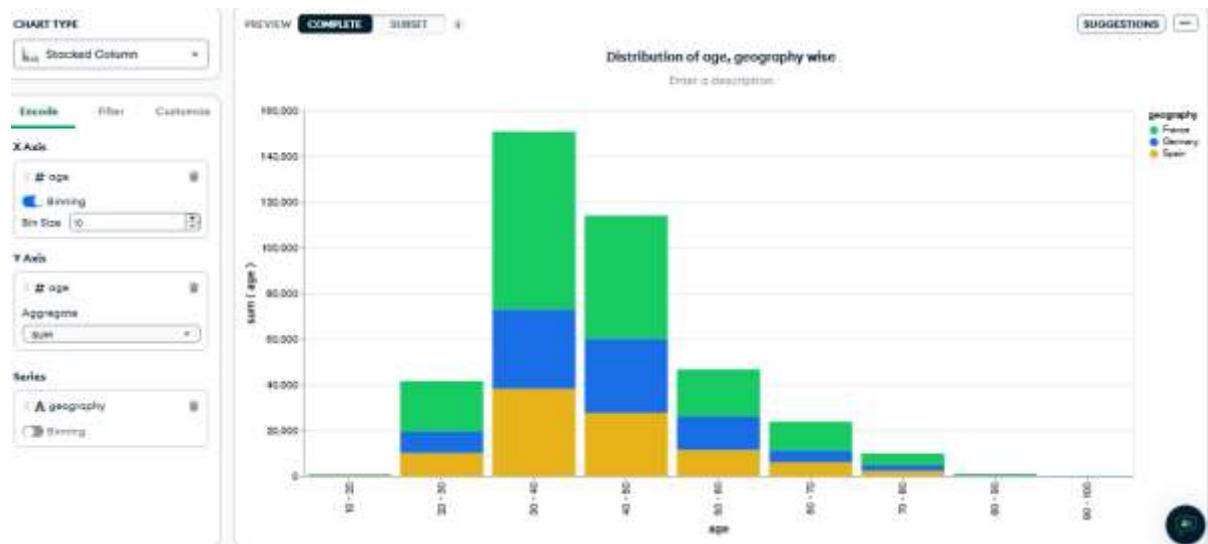


Figure 19: Green has the highest proportion in relation to the total.

And the corresponding stacked column chart is below (color palette is the same as above):



14. How to change legend colors?

See this [link here](#) and also [this link](#).

To change Legend color, click on *Customise* tab and select *By Series*. Change palette as you please.

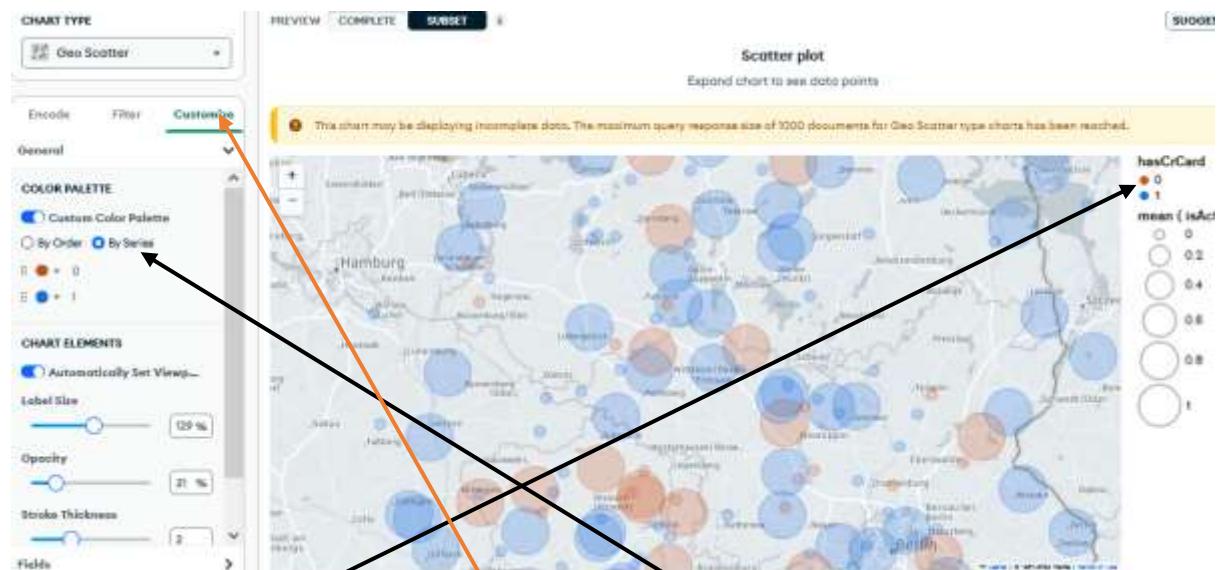


Figure 20: To change Legend color, click on *Customise* tab and select *By Series*. Change palette as you please

15. Import/export a Dashboard

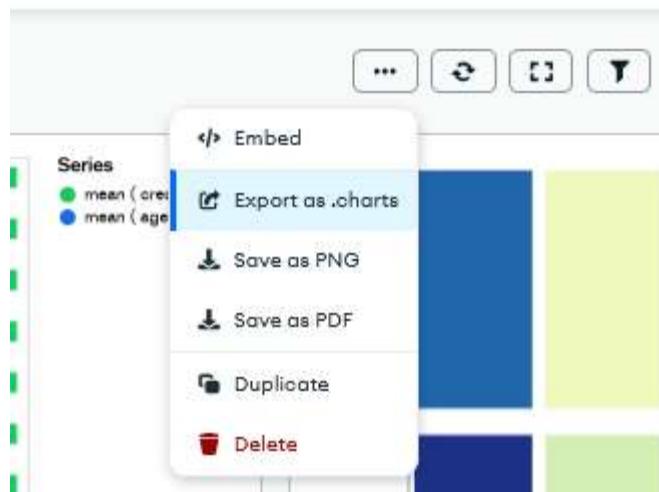
(see [link here](#))

Here is how to export and import a dashboard.



Figure 21: To export, click on three-dots and then click on **Export as charts**. Dashboard is saved as: **churnModeling.charts**

An expanded view of a section of above is as follows:



To import a dashboard, click on the down arrow besides **Add Dashboard** and click **Import dashboard**. Select the previous exported file.

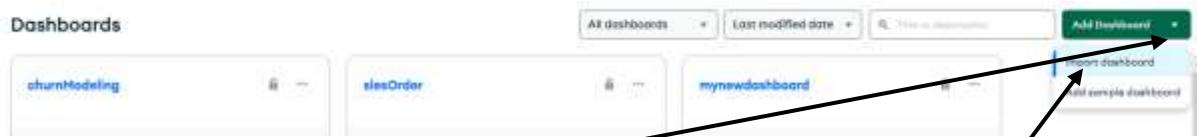
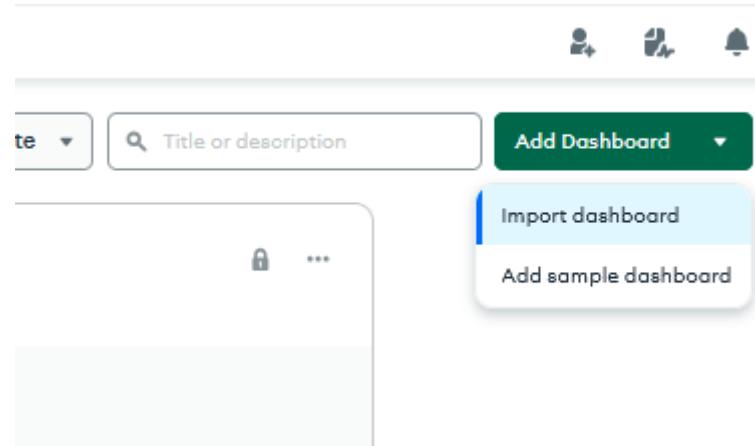


Figure 22: To import a dashboard, click on the down arrow besides **Add Dashboard** button and click **Import dashboard**. Select the previously exported file.

An expanded view of a section of above is as follows:



16. Refresh for streaming data

(see [link here](#))



Figure 23: At the top-right of the dashboard, click on refresh icon and then click on Refresh settings

An expanded view is as below:



On clicking Refresh settings, the following dialog box opens:

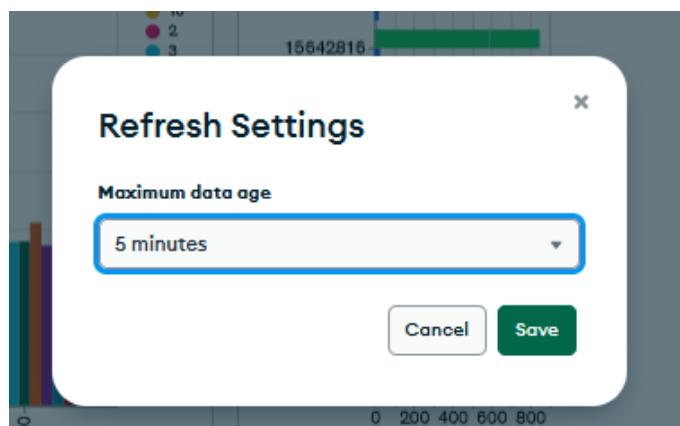


Figure 24: Set your Refresh settings and click Save.
