

Data Visualization with Tableau



Business Intelligence (Tableau)

1.1 Introduction:

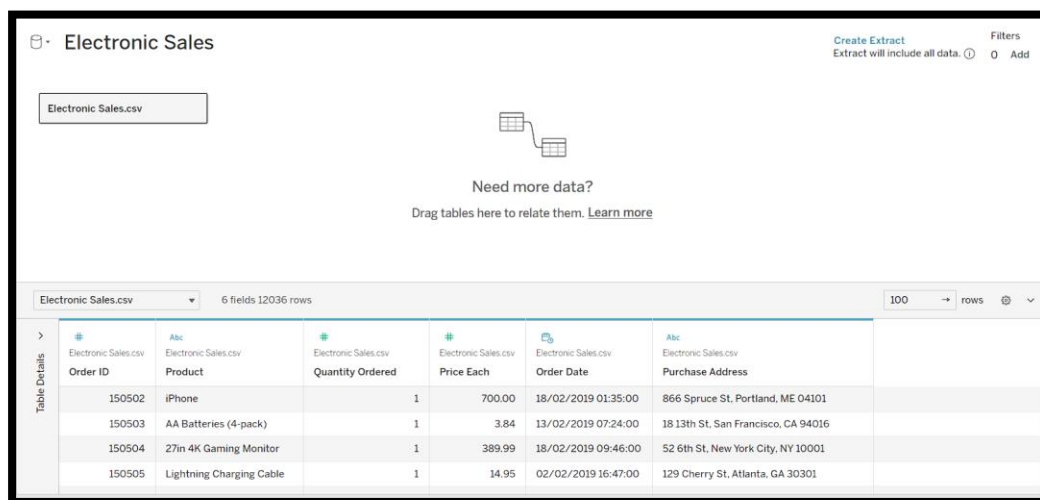
Tableau is a popular data visualization tool data analysts and organizations use to create interactive and visually appealing dashboards and reports. Its popularity can be attributed to its ability to provide insights and discoveries into data. Tableau is extremely effective for data visualization through its simple data tables and dazzling visualizations.

1.2 Problems descriptions:

This report aims to visualize through deep analysis the "Electronic Sales" dataset obtained from Walmart, one of the leading supermarkets in the USA. As data Analysts of the R&D of Amazon, we were assigned to analyze the dataset using Tableau and complete several tasks for completing the assignment of our academic modules Big Data Analytics. The dataset contains important information such as the product ID, quantity ordered, product name, price, order date, and address. The main objective of this report is to provide valuable insights into the dataset by completing various tasks using Tableau. The report presents the analysis of valuable findings along with the necessary visualizations and explanations.

1.3 Data description:

This data set has six fields (columns) and 12036 rows. Data set has the sales record of February month in 2019. Here is the first view of the dataset after loading it into Tableau.



Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address
150502	iPhone	1	700.00	18/02/2019 01:35:00	866 Spruce St, Portland, ME 04101
150503	AA Batteries (4-pack)	1	3.84	13/02/2019 07:24:00	18 13th St, San Francisco, CA 94016
150504	27in 4K Gaming Monitor	1	389.99	18/02/2019 09:46:00	52 6th St, New York City, NY 10001
150505	Lightning Charging Cable	1	14.95	02/02/2019 16:47:00	129 Cherry St, Atlanta, GA 30301

Figure 1: First Look of "Electronic Sales" data in Tableau.

Task 1- Ten Top-Selling and Ten Low-Selling Products:

To find this we need to find the most ordered and least ordered products based on the Quantity Ordered field. To do that first, we will take the "products" to the columns field and then we will take the "Quantity ordered" to the row field. Now we get all product and their sum of quantity ordered.

Figure 2: Filtering 10 top products

Then need to arrange as descending from the side of the chart. From the mark pen we can design the color, label by dragging and customizing the fields.

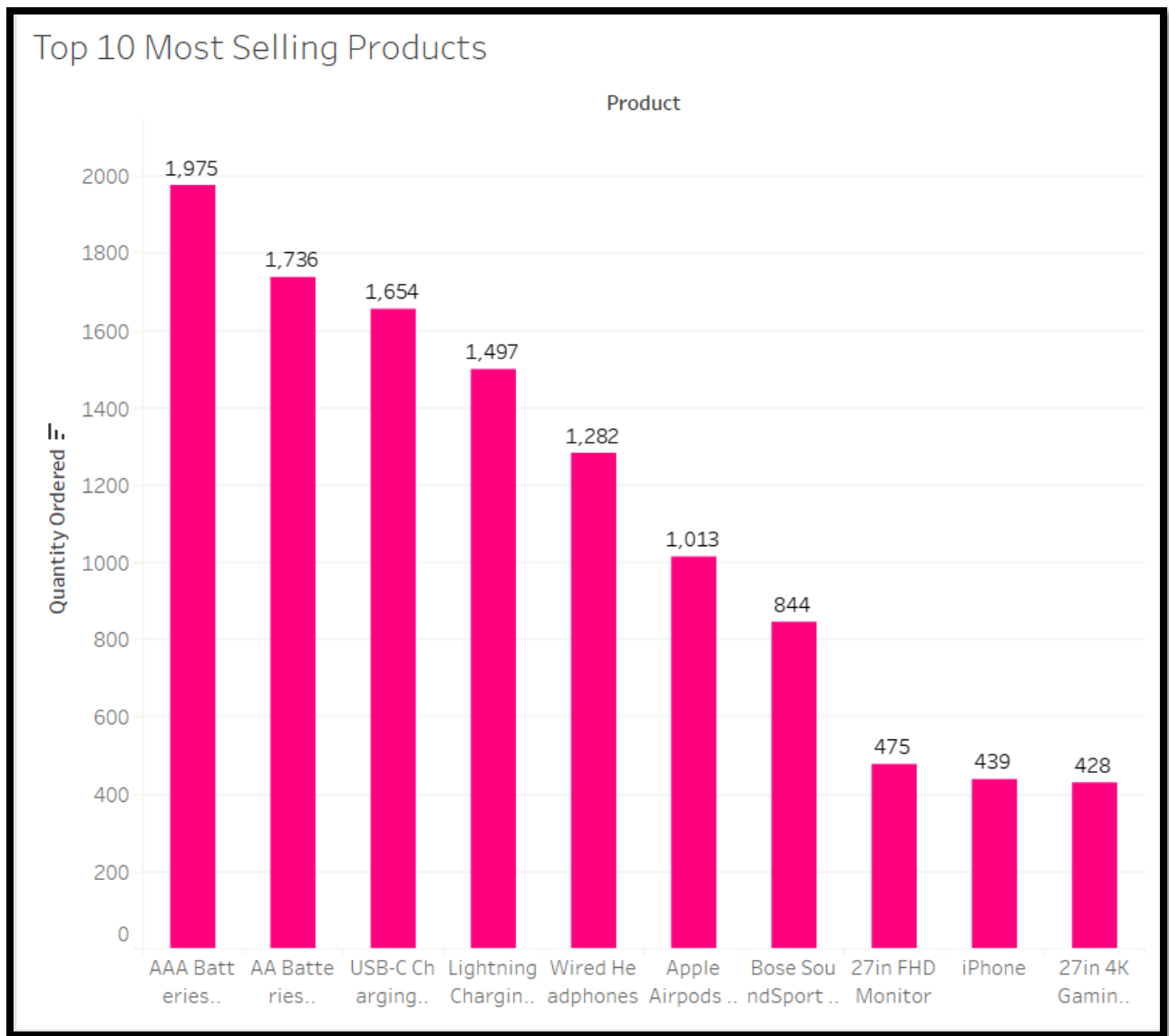
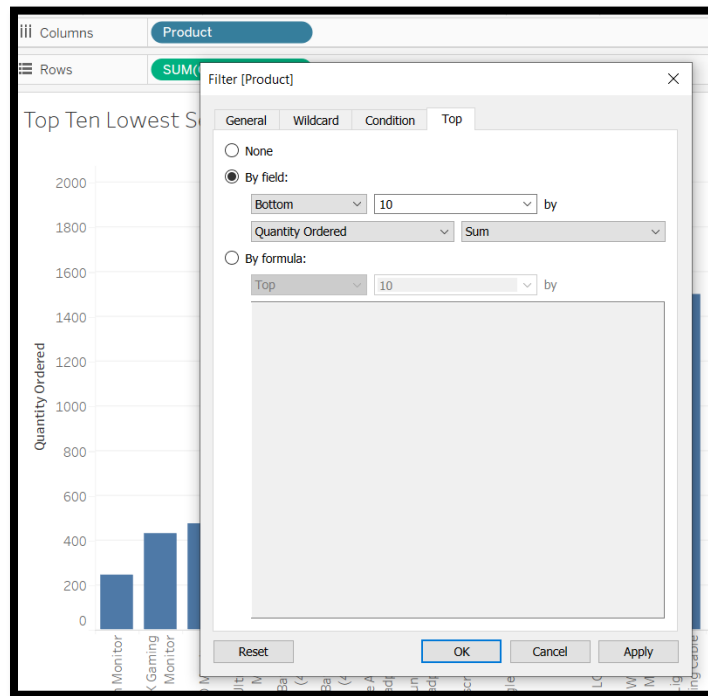


Figure 3: Top ten most selling products



This chart shows that AAA Batteries is the best-selling product with the number of 1975 quantity ordered.

Now in another worksheet, in the same way we found the top ten lowest selling products.

Figure 4: Filtering Bottom Ten Selling Products

Then we changed the color, label, and size from the mark pen. We need to drag the quantity ordered in the label field; it shows the sum of the total quantity ordered of those products.

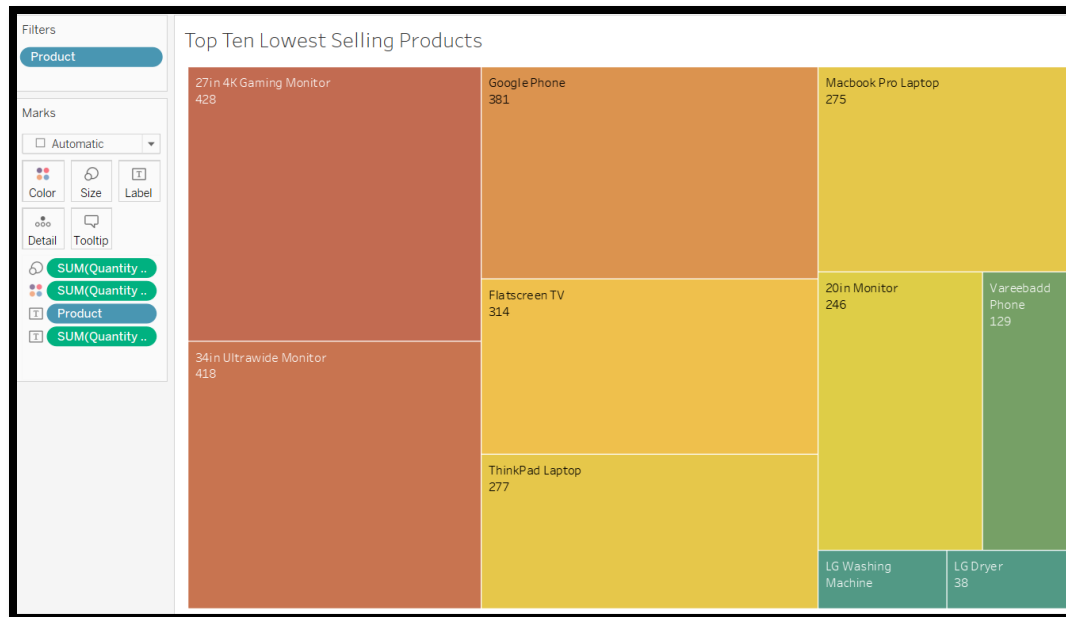


Figure 5: Area Plot of Top 10 Lowest Selling Products

Here we see that the lowest ordered product was LG dryer. Only ordered 38 pieces, whereas LG washing Machine was in the second position of lowest selling list.

Task 2 - Total/Average Sales Made in Each City:

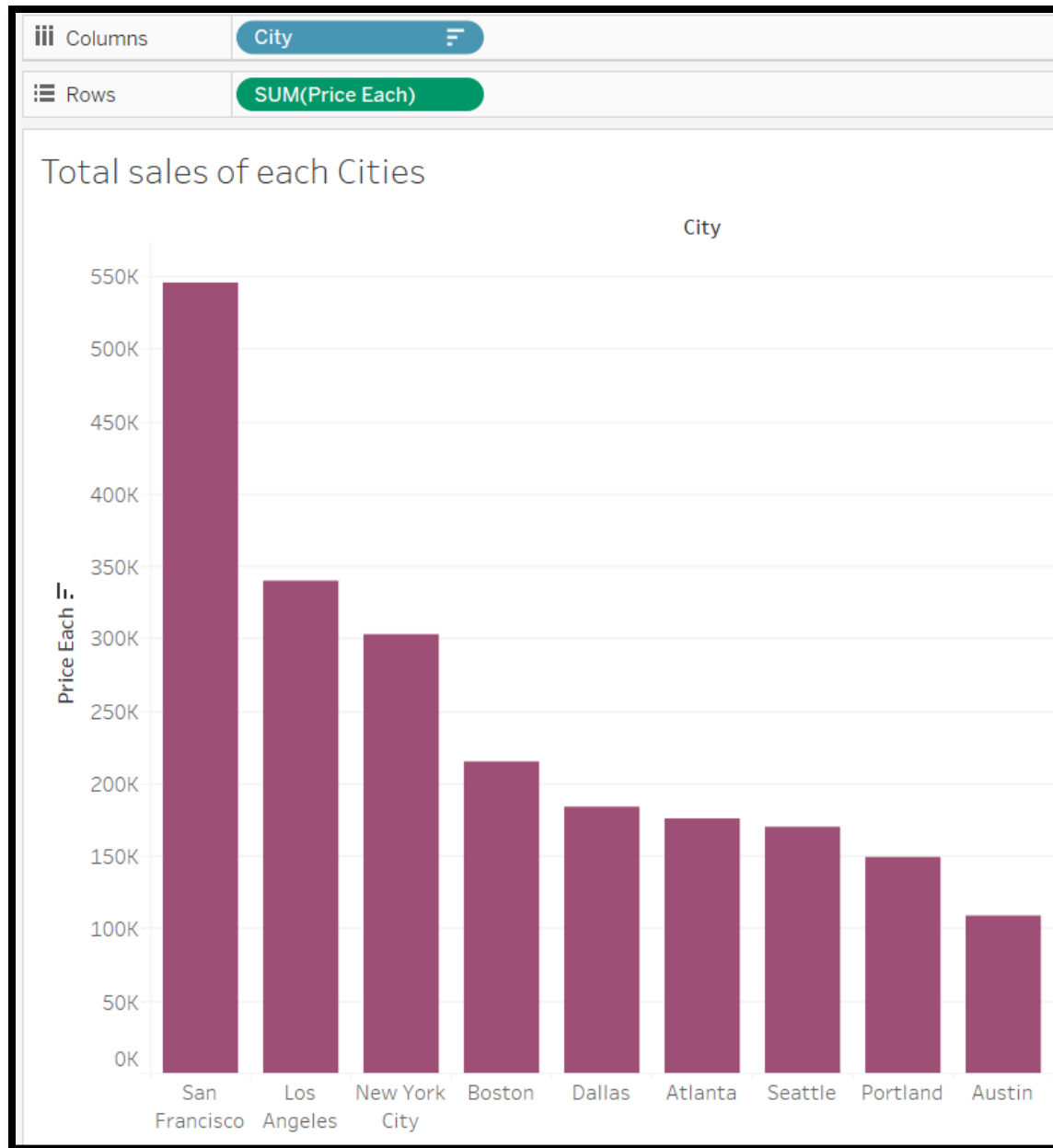
Our data do not have any field by the name city. To calculate sale by city we need to create a calculative field by the name "City". To do that, first, we duplicated the address field which creates another field. Now we need to split that new column by delimiter (,). It then again created 3 new fields; we kept the various city name 'the middle' part by renaming them as 'City' and then deleted the other two. Now our data have a new field name, 'City'.

Then I rename this 'City with Address'.

100 → rows ⚙️				
Electronic Sales.csv	Electronic Sales.csv	Calculation	Calculation	Calculation
Order Date	Purchase Address	City with Adress	City	Warranty End Date
2/18/2019 1:35:00 AM	866 Spruce St, Portland, ME ...	866 Spruce St, Portland, ME ...	Portland	8/18/2019
2/13/2019 7:24:00 AM	18 13th St, San Francisco, CA ...	18 13th St, San Francisco, CA ...	San Francisco	8/13/2019
2/18/2019 9:46:00 AM	52 6th St, New York City, NY ...	52 6th St, New York City, NY ...	New York City	8/18/2019
2/2/2019 4:47:00 PM	129 Cherry St, Atlanta, GA 30...	129 Cherry St, Atlanta, GA 30...	Atlanta	8/2/2019
2/28/2019 8:32:00 PM	548 Lincoln St, Seattle, WA 9...	548 Lincoln St, Seattle, WA 9...	Seattle	8/28/2019
2/24/2019 6:50:00 PM	387 12th St, Austin, TX 73301	387 12th St, Austin, TX 73301	Austin	8/24/2019

Now I need to Drag the "City" field from the "Data" pane to the "Columns" shelf. And then create another calculative field named "Sales". Click on the "Analysis" menu at the top of the screen, we selected the "Create Calculated Field", and created a new field called "Total Sales" with the formula: ([Price Each] *[Quantity Ordered]). Now we have "Sales" column.

Steps are as follows: -



Then by Clicking on the drop-down menu next to the "Total Sales" field name in the "Marks" card and selecting "SUM" or "AVG" to switch between sum and average sales. By dragging the sales again in rows field and selecting SUM we can get two comparative sales of the different city.

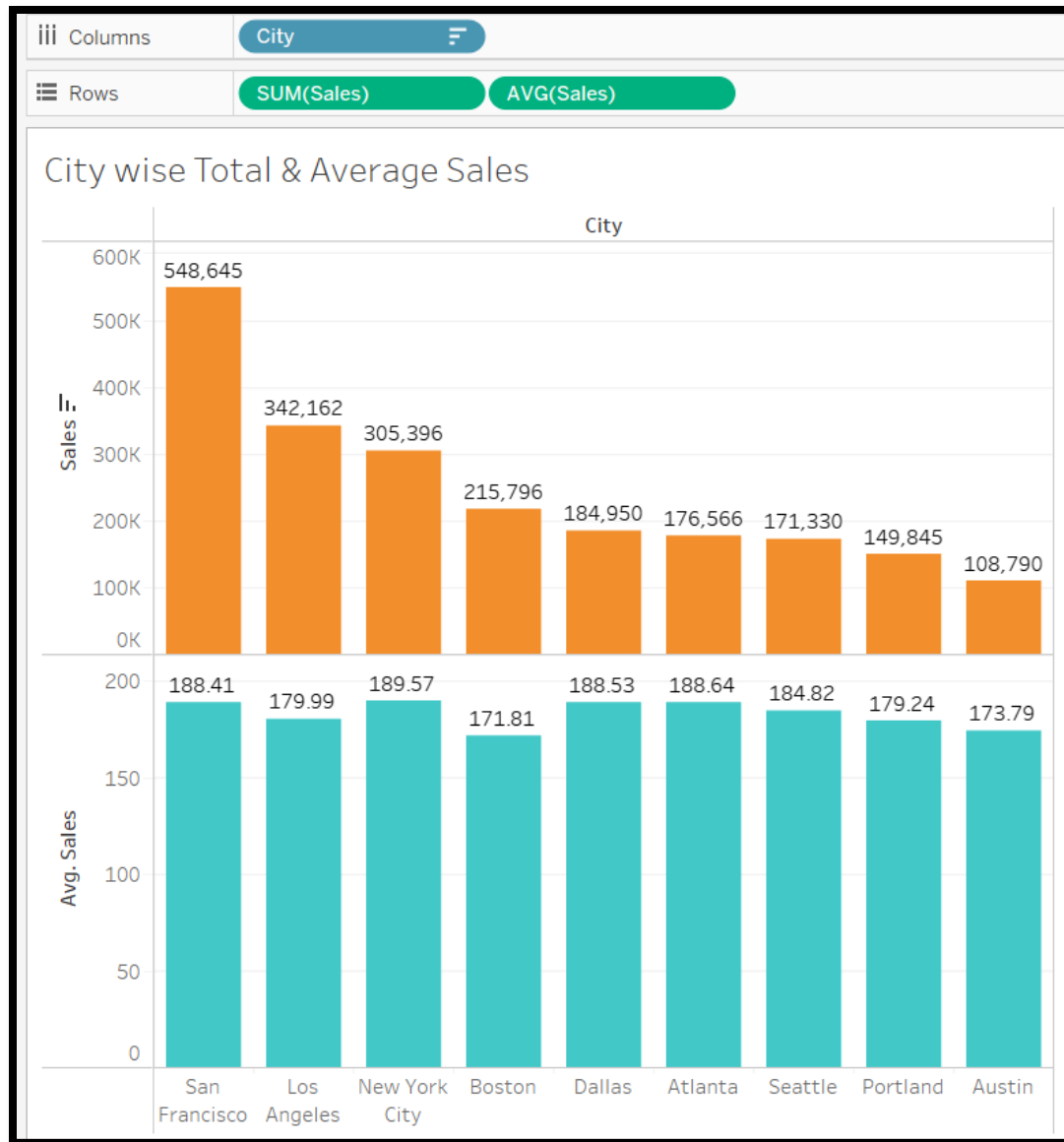


Figure 6: City Wise Total and Avg. Sales

We see that top selling city was San Francisco, where as Austin was in lowest position. But in average sales New York City was in best position, in contrary Boston was bottom average selling.

Task 3 - Stores' Performance on Weekly Basis

First, we need to create parameter. We set it as “weekly Sales”, we kept allowable values list and then put week, which will be display as Weekly sales.

The screenshot shows the 'Edit Parameter [Weekly Sales]' dialog box. The background shows a worksheet named 'Sheet 10' with a 'Drop field here' area. The dialog box contains the following fields and options:

- Name:** Weekly Sales
- Properties:**
 - Data type:** String
 - Display format:** Week
- Current value:** Weekly Sales
- Value when workbook opens:** Current value
- Allowable values:**
 - ☐ All
 - ☒ List
 - ☐ Range
- | Value | Display As |
|--------------|--------------|
| Week | Weekly Sales |
| Click to add | |
- ☒ Fixed
- ☐ When workbook opens
- Add values from ▼
- Remove Selected
- Cancel
- OK

Figure 7: Weekly Sales Parameter Screen Shot.

Then we created a calculated field, 'Date Truncate Calc' by using the formula DATETRUNC ('week', [Order Date]).

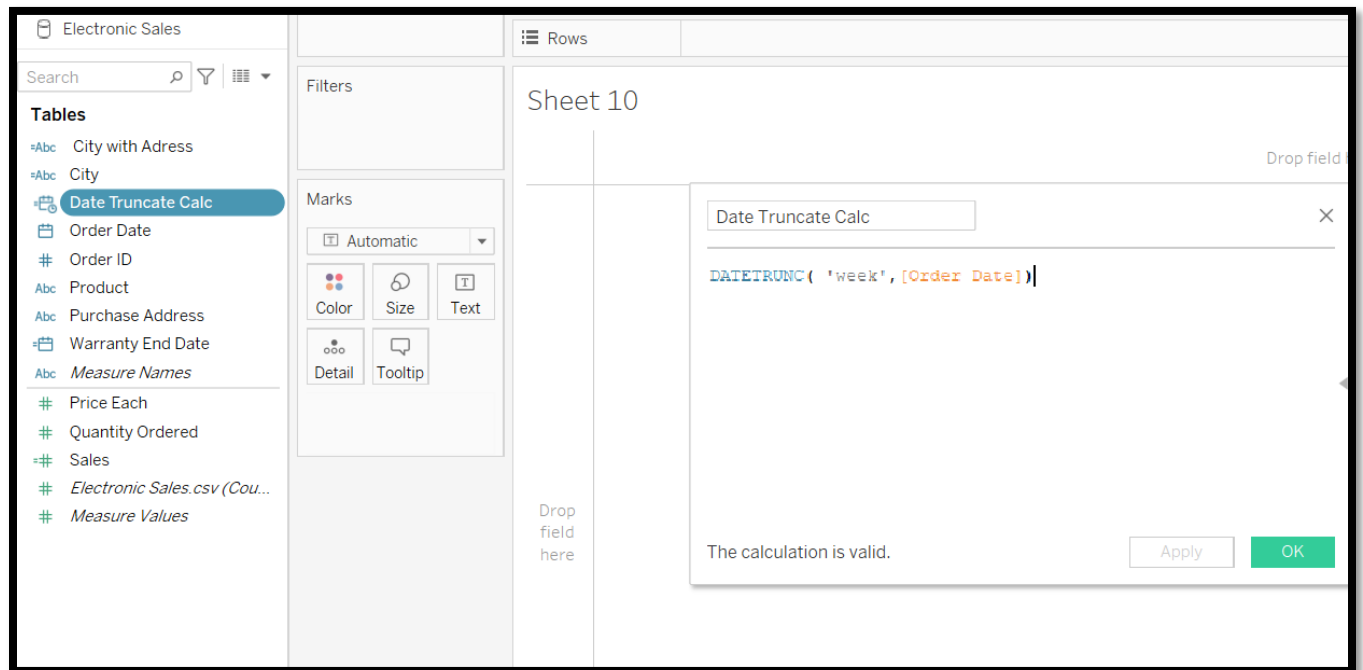


Figure 8: Calculation of Date Truncate with Formula

Then we need to drag the date truncate calc in Column field. Then from the drop-down menu we can choose the number of weeks. After that, we will set SUM(Sales) and AVG(Sales) in row field. Consequently, we can get the following comparative chart of Sum of Sales and average Sales.

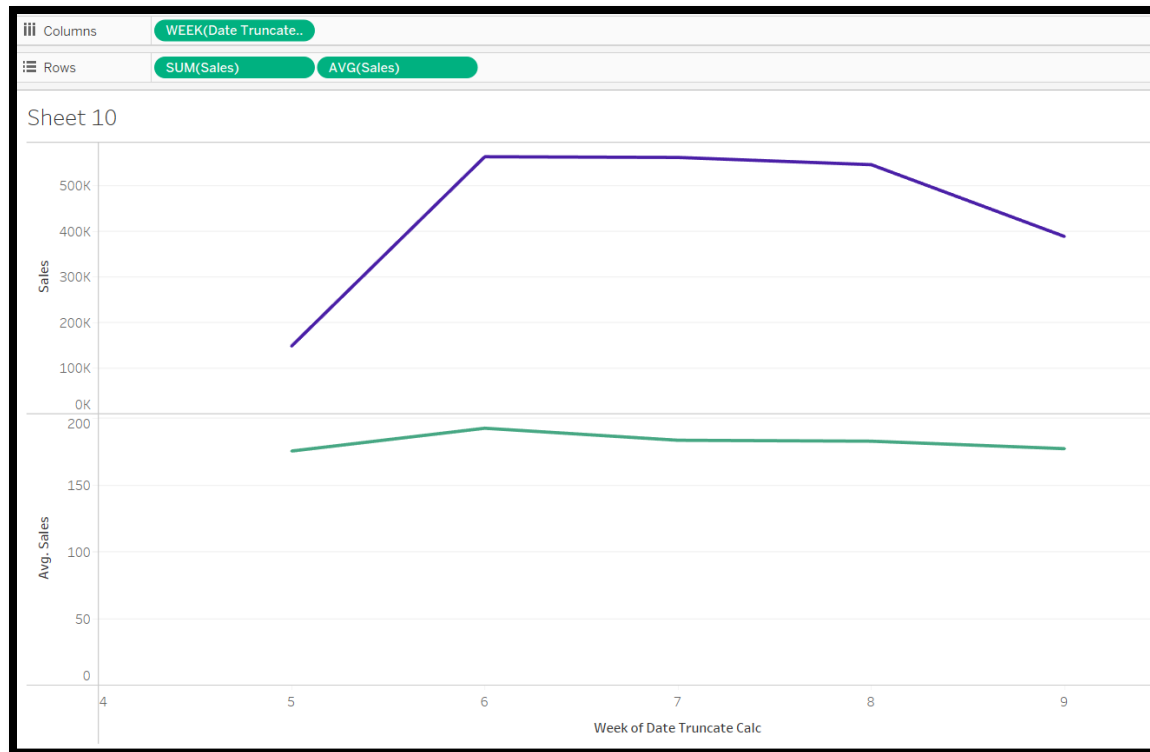


Figure 9: Line Chat of Weekly Sales as Number of Week

In our data, had the sales record from end of the January to end of the February 2019. S, we can see the sales record from fifth week to week no 9.

Whereas we can also set the date parameter in selecting the weekly sale, from our 'Date Truncate Calc'. From the daily sales, we need to select weekdays. Then it will give us more clear weekly sales from 27 January to 24 February of 2019.

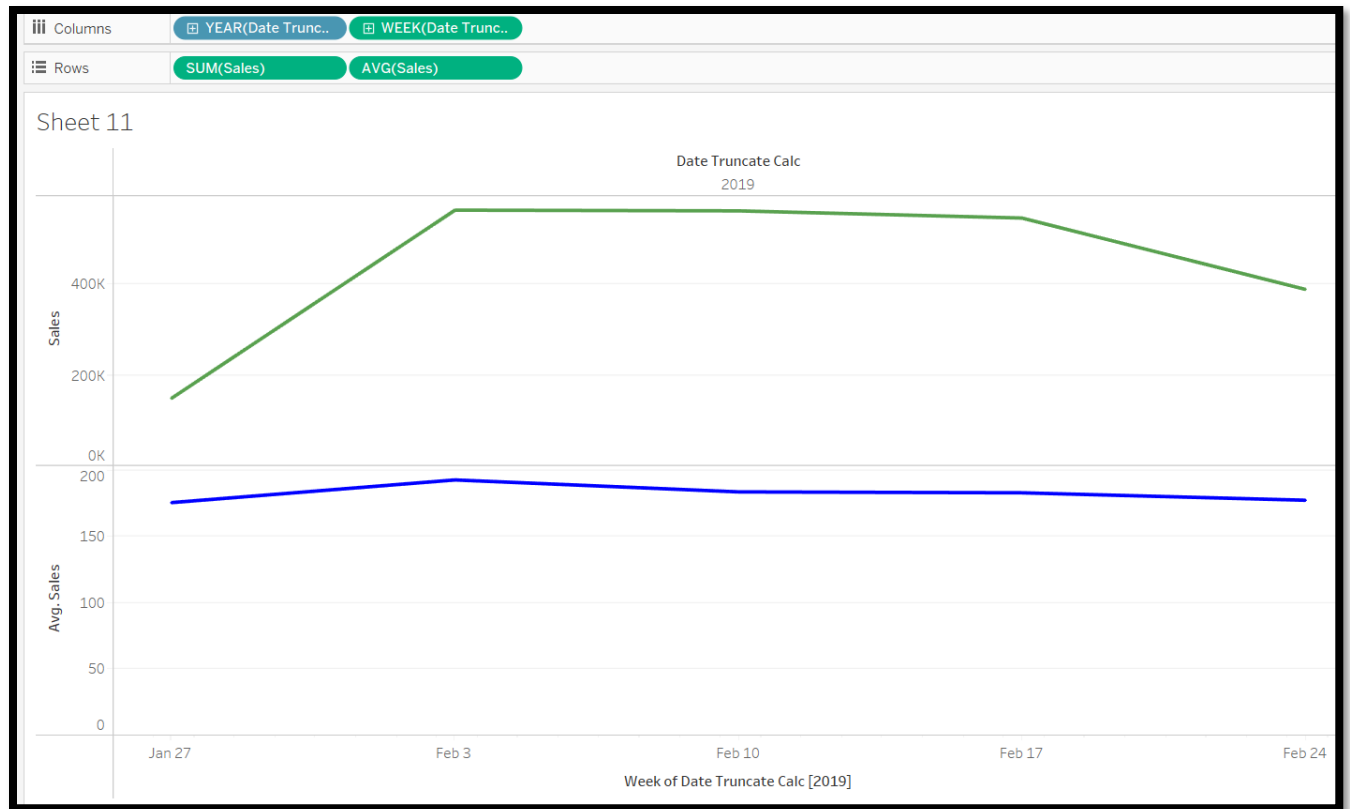


Figure 10: Weekly Sales, From Date to Date

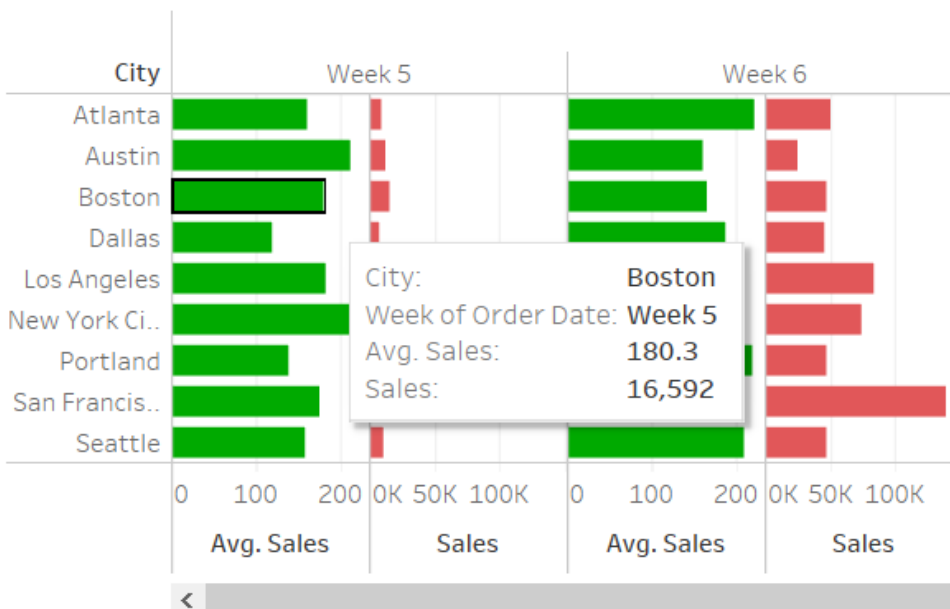
Here, there was an sharp increase of weekly sales of 27th January to 3rd February in both SUM(Sales) and AVG(Sales). After that Average Sales was constant whereas Some of sales were dropped in the last week (17th Feb to 24th Feb).

Weekly Store Performance by City:

Now we want to visualize the weekly Store's performance in different City for more comparative analysis by putting the



Weekly stores' performance



If we over on any city, it will show the weekly Avg. and Total sales of that City.

Task 4 - Calculating the Warranty End Date:

To create a calculated field by the name “Warranty End Date”, in the formula editor, we use the DATEADD function to add 6 months to the purchase date field. The syntax for the function is as follows.

DATEADD ('month', 6, [Order Date])

The screenshot shows a formula editor window titled 'Warranty End Date'. The main text area contains the formula `DATEADD('month', 6, [Order Date])`. To the right, there is a search dropdown menu with 'All' selected and a list of functions including ABS, ACOS, AND, AREA, ASCII, ASIN, ATAN, ATAN2, ATTR, AVG, and BUFFER. At the bottom, there are buttons for 'Apply', 'OK', and 'Apply and Close'. A status message at the bottom left says 'The calculation is valid.'.

Figure 11: Creation of Warranty End Date

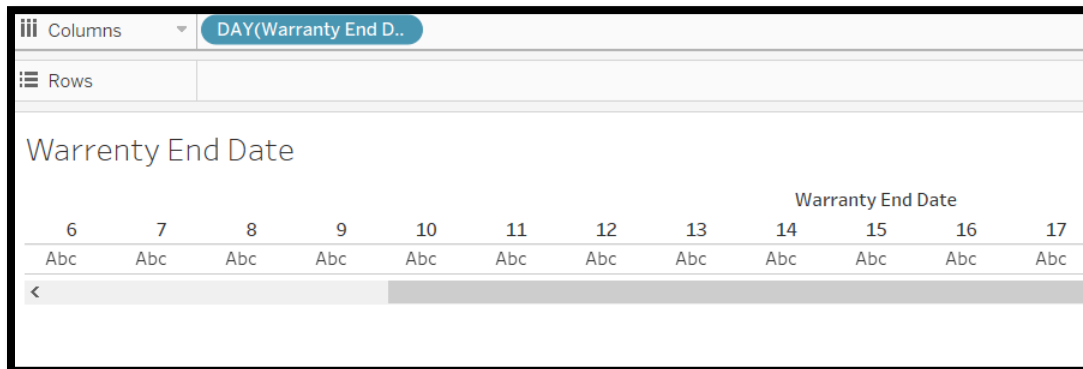
Now we have a warranty end date.

The screenshot shows a data source table with 5 columns: 'Order Date', 'Purchase Address', 'City with Address', 'City', and 'Warranty End Date'. The 'Warranty End Date' column is highlighted in blue. The table contains 6 rows of data.

Order Date	Purchase Address	City with Address	City	Warranty End Date
3/1/2019 1:03:00 AM	683 Adams St, Portland, OR ...	683 Adams St, Portland, OR ...	Portland	9/1/2019
3/1/2019 3:06:00 AM	270 Dogwood St, San Francis...	270 Dogwood St, San Francis...	San Francisco	9/1/2019
2/28/2019 1:55:00 PM	579 9th St, Atlanta, GA 30301	579 9th St, Atlanta, GA 30301	Atlanta	8/28/2019
2/28/2019 8:32:00 PM	548 Lincoln St, Seattle, WA 9...	548 Lincoln St, Seattle, WA 9...	Seattle	8/28/2019
2/28/2019 6:16:00 PM	457 Maple St, Los Angeles, C...	457 Maple St, Los Angeles, C...	Los Angeles	8/28/2019
2/28/2019 1:47:00 PM	311 8th St, Austin, TX 73301	311 8th St, Austin, TX 73301	Austin	8/28/2019

Figure 12: Picture of Warranty End Date column in data Source

- Now we drag the Warranty End Date field to your visualization to see the warranty end date for each ordered item.



We set the Product in row field.

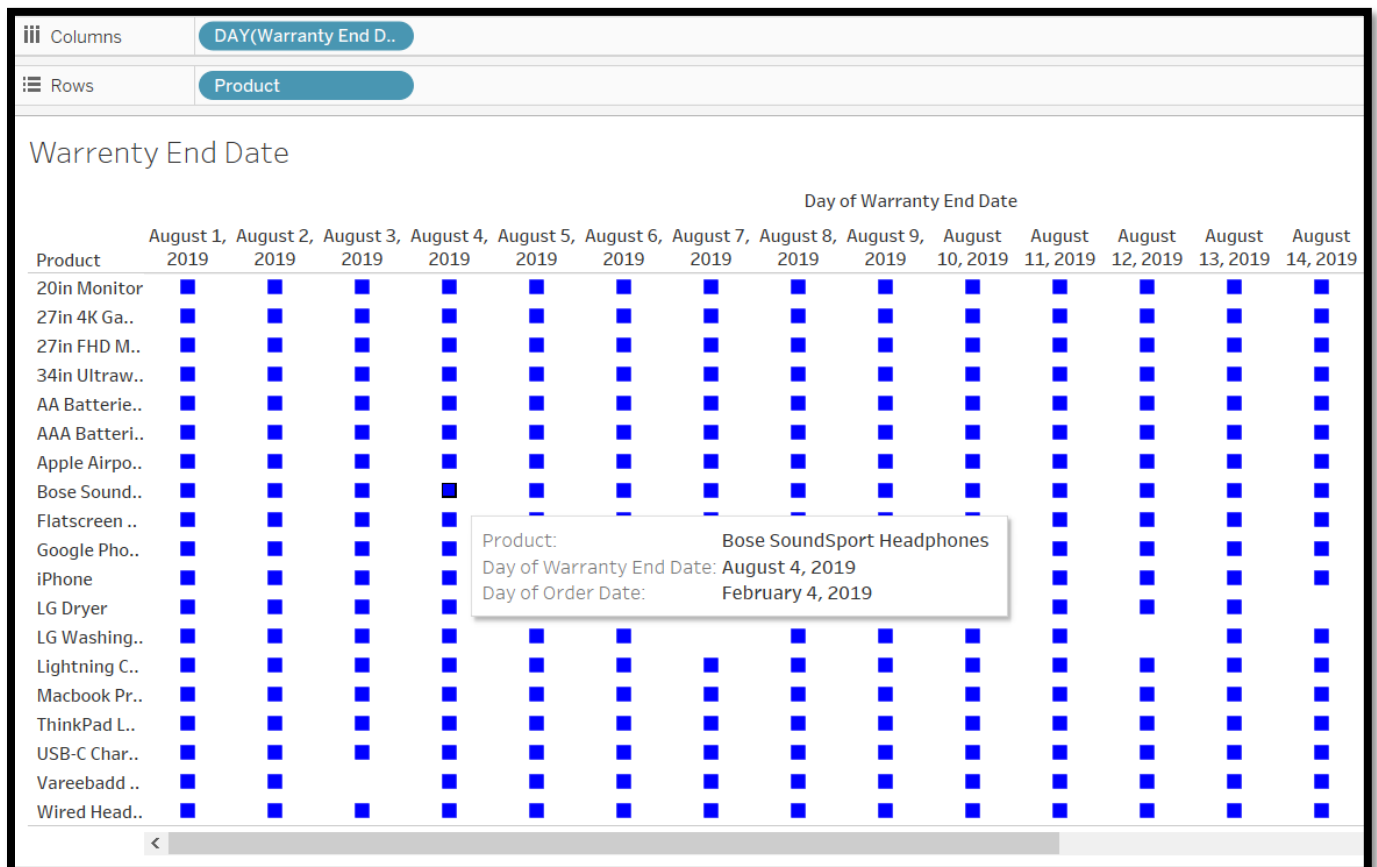


Figure 13: Visualization of Product's Warranty End Date

As there are so many data, we cant see the every thing visually at a glance. But if we hover on the chart, every 'order date' and 'warrantee' end date with product name is clearly visible here.

Task 5 - Creating Interactive Dashboard of Electronic Sales:

Step 1: From the top menu bar, by hitting the dashboard button and we can see a dashboard canvas. Then, all our independent worksheets need to be added into dashboard just by double clicking on the name of the worksheet on the left side of the dashboard.

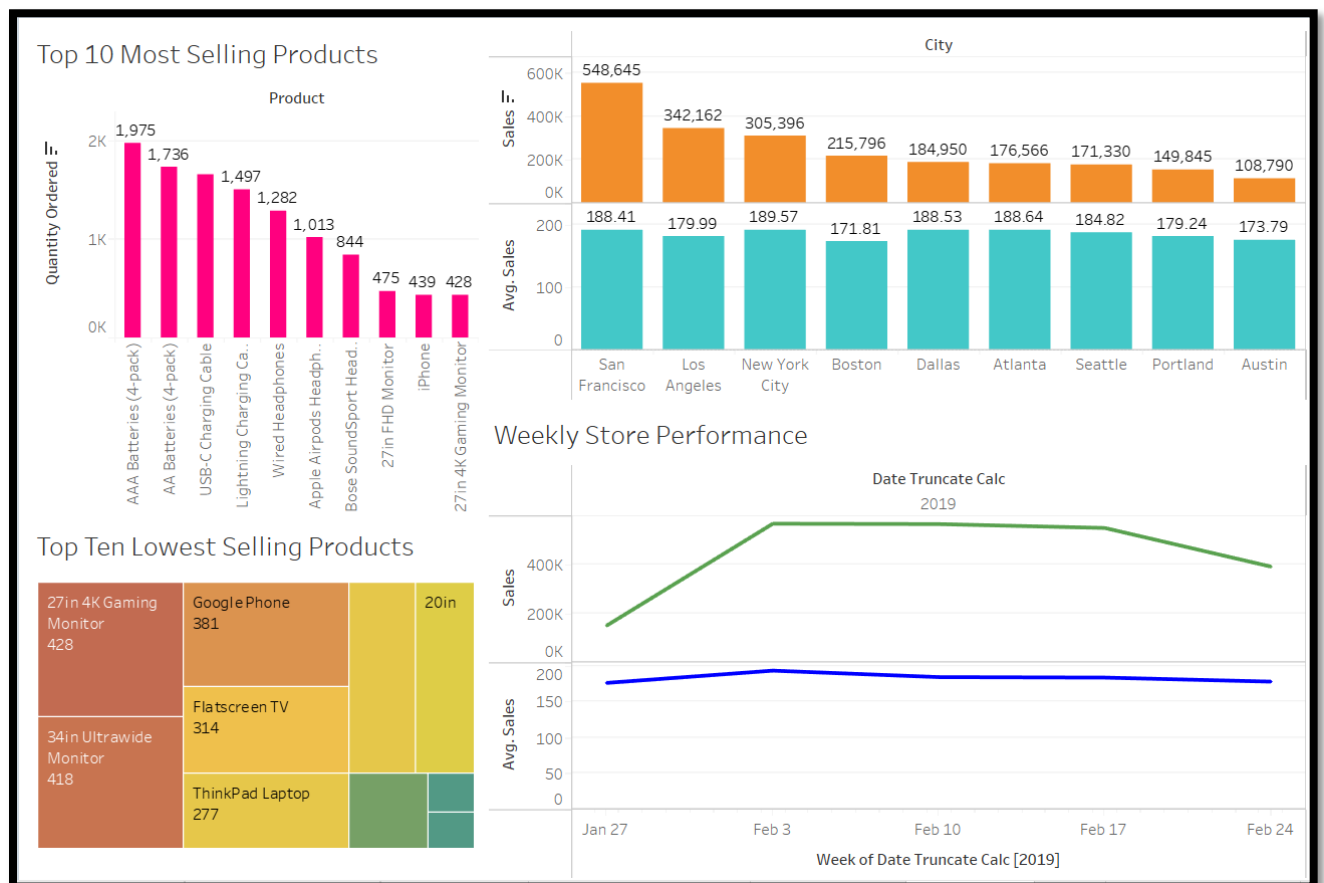
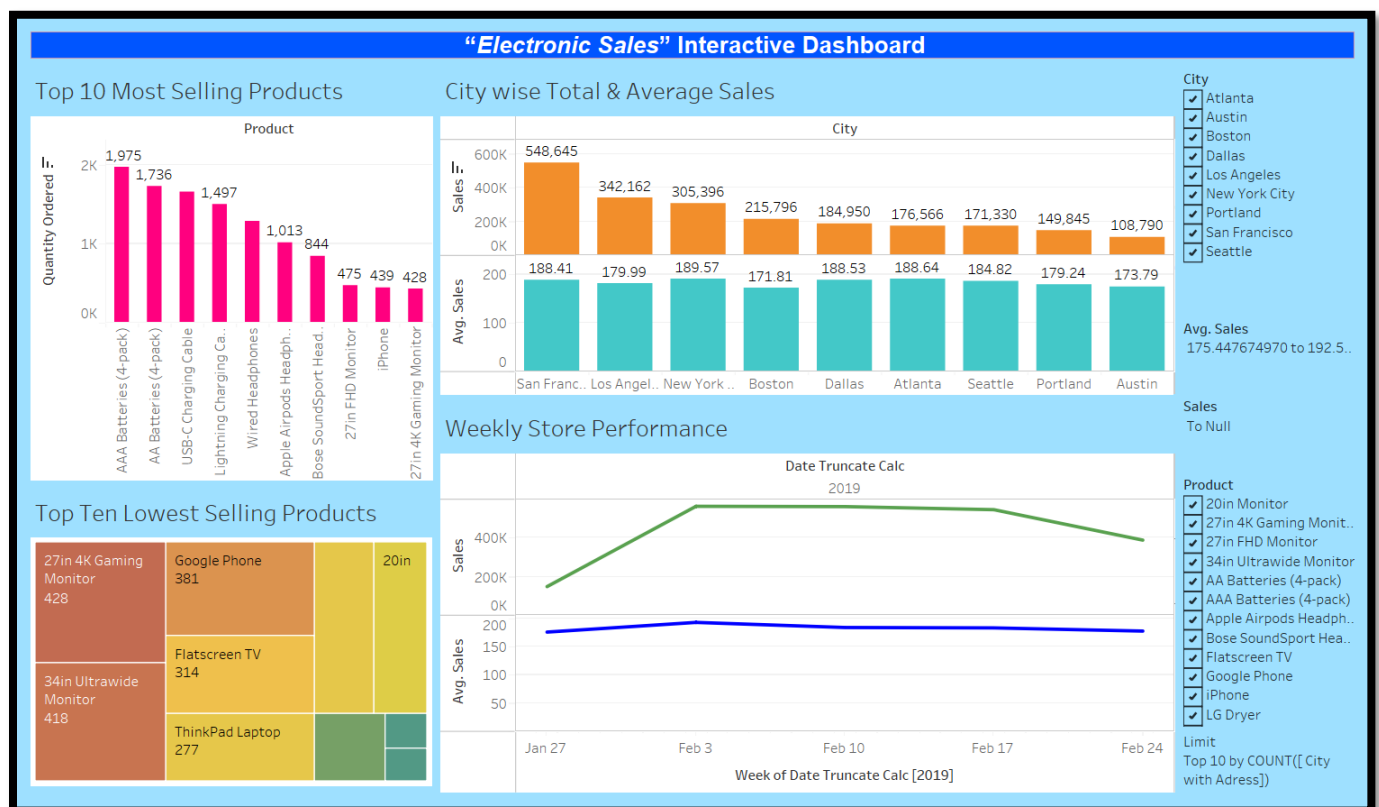


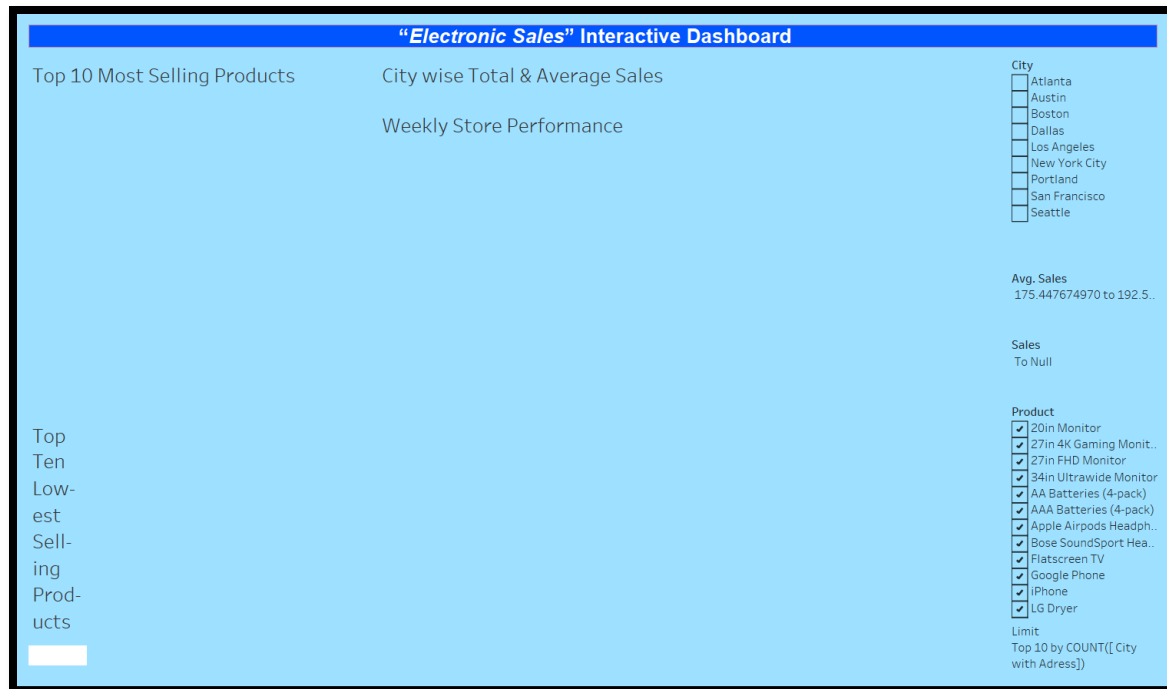
Figure 14: Picture of Dashboard Just After Adding the Sheets

Step 2: Then we have customized the dashboard by using the formatting tools to customize the layout, size, and orientation of the dashboard. Then we added titles, text boxes, and the background color of the dashboard, which is extremely useful to explain the data and visualizations.

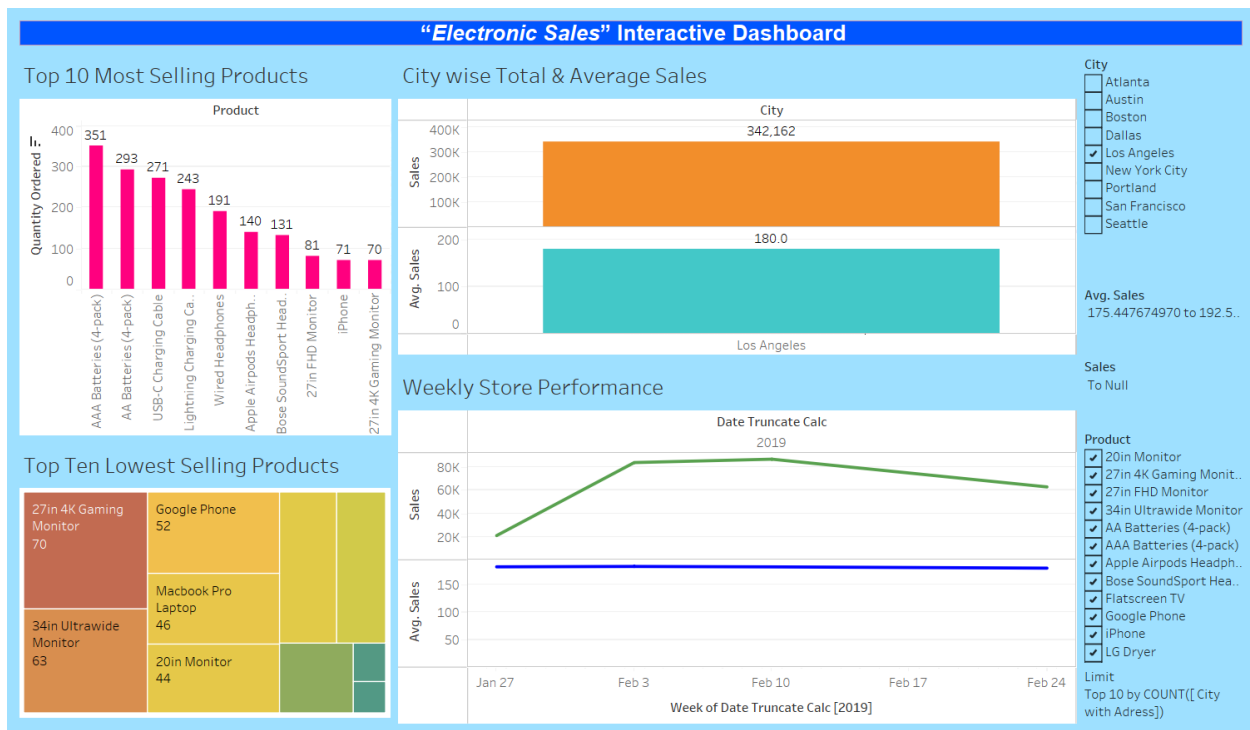
Step 3: To make our dashboard interactive, we have set the filters and drop-down menus. Then we created interactive visualizations by using the "Actions" feature to link different worksheets in this dashboard.



Now our all four worksheets are interrelated with each other. If we unmark the tick from all it will show nothing.

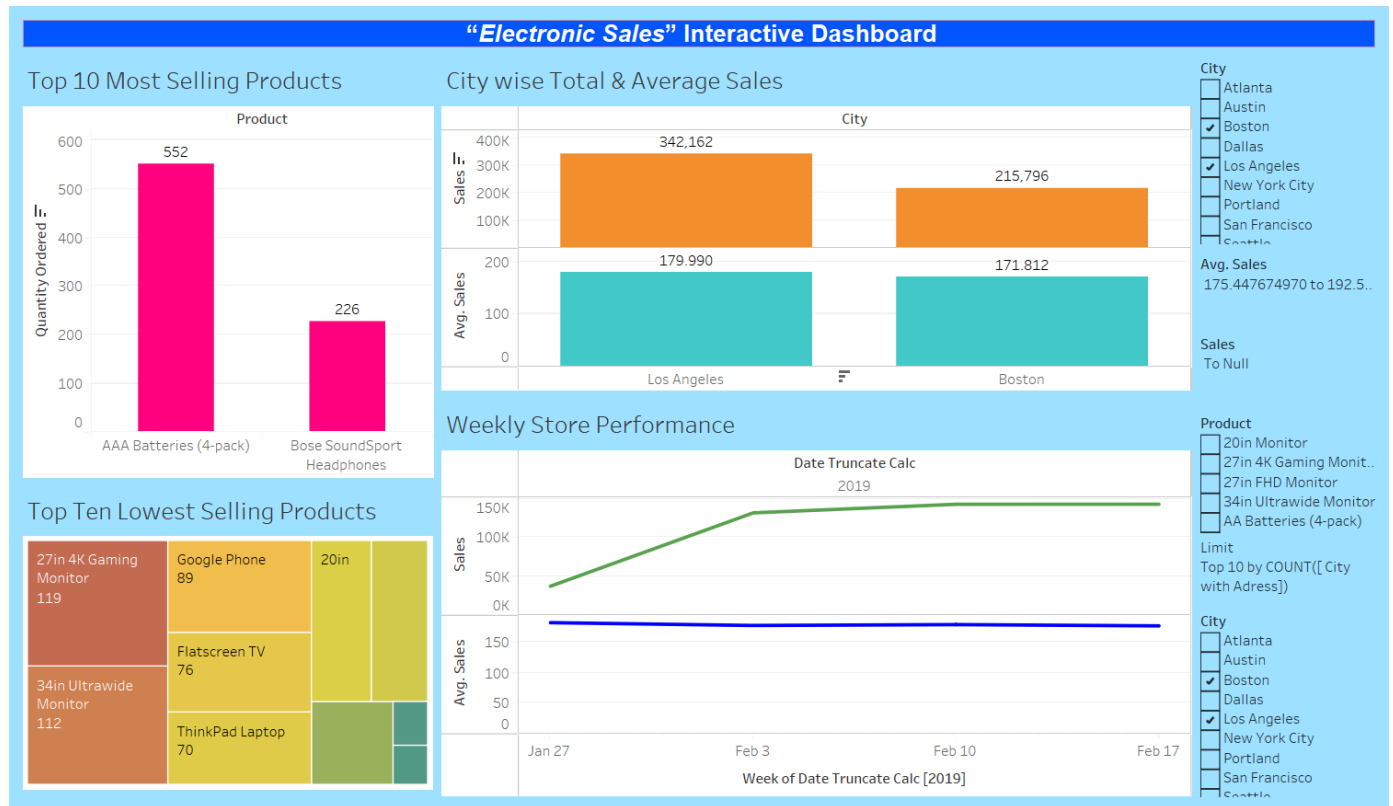


Then, now we will check the sales on basis of Los Angeles City by clicking on that particular city.



Here we can only see the sales of Los Angeles City.

Together with the city we can also filter by specific product. Here is the result of two products sales in two City.



Finally, we can present the insight with any parameter in this interactive dashboard.

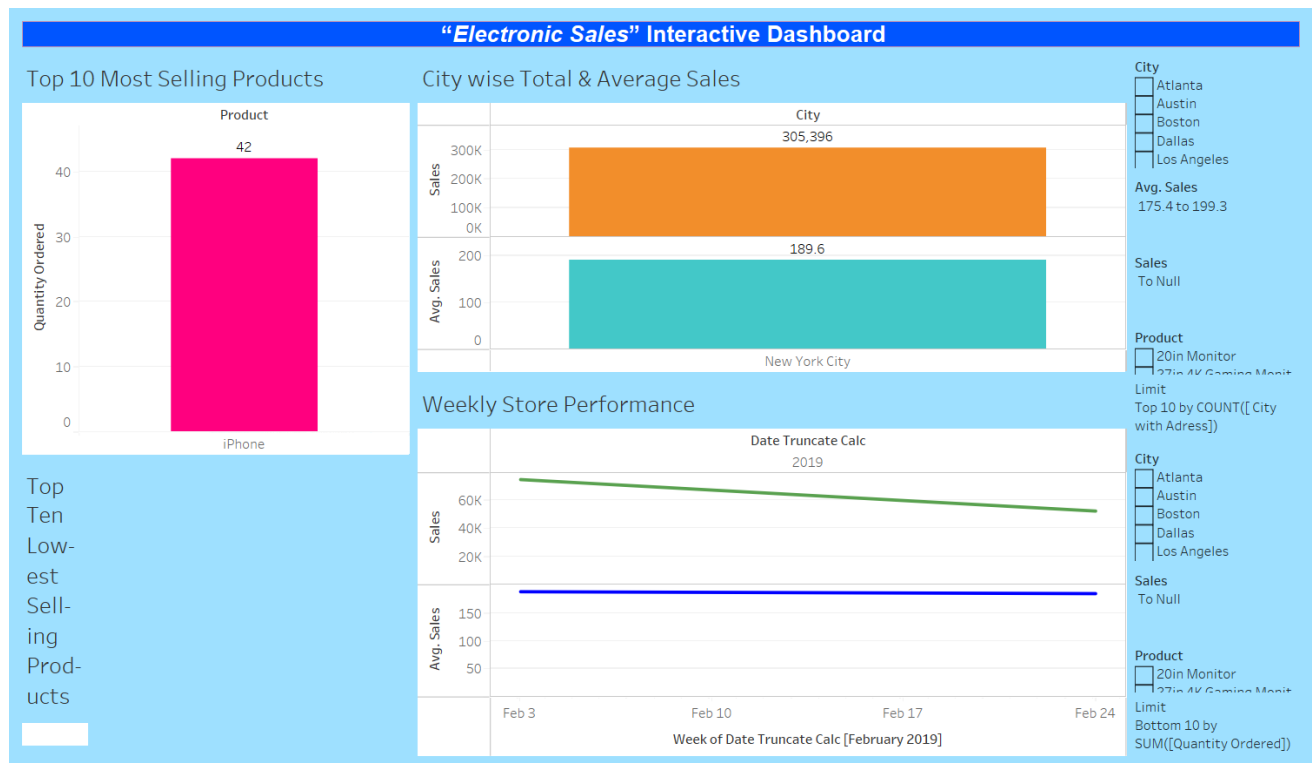


Figure 15: Picture of Interactive Dashboard

Here is the New York City's total and average sales report along with weekly performance trends.

Conclusion:

Data visualization is a complex process that has potential applications in diverse disciplines. The future of data visualization is being shaped by emerging sources of intelligence in the architecture of Tableau. However, we need to remember that effective data visualization should be substantive and accurate, and the dashboards need to be interactive for connecting to other data to get valuable insight into the data.