Fundamentals of Data Visualization Final Project

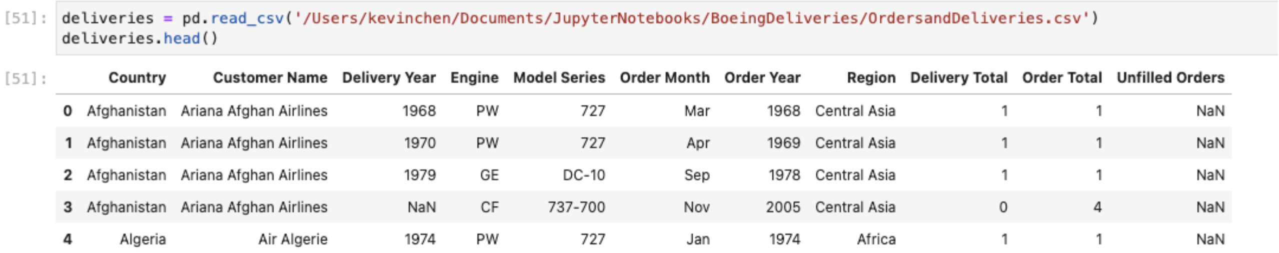
Dataset

For this final project, I will be using a dataset from kaggle that has information on every commercial jet that Boeing has delivered since 1958. The link to the dataset is https://www.kaggle.com/datasets/nurielreuven/boeing-historical-airplane-orders-deliveries?resource=download.

There are several different columns in this dataset. The columns are:

- Country: The country that the airline is headquartered in
- Customer Name: The airline that is receiving the aircraft
- Delivery Year: The year that that aircraft was delivered to the airline customer
- Engine: The company that built that engine for the given aircraft since Boeing does not manufacture the engines.
- Model Series: The model of the aircraft followed by the aircraft configuration
- Order Month: The month that the aircraft was ordered on
- Order Year: The year the the aircraft was ordered on
- Region: What part of the world the customer airline is located
- Delivery Total: How many of the orders in a contract order are fulfilled
- Order Total: The total number of orders in a contract
- Unfulfilled Orders: The number of aircraft orders that have not been fulfilled

Below are a few rows of the dataset as an example.



```
[51]: deliveries = pd.read_csv('/Users/kevinchen/Documents/JupyterNotebooks/BoeingDeliveries/OrdersandDeliveries.csv')
      deliveries.head()
```

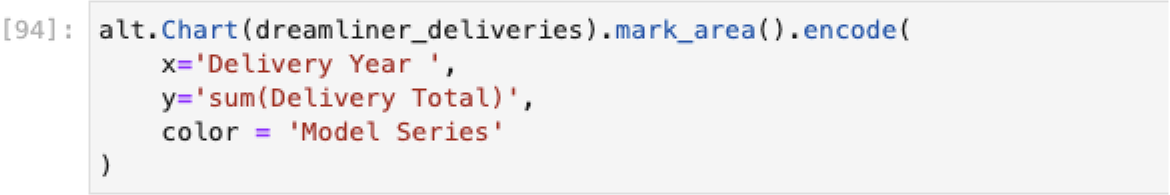
[51]:

	Country	Customer Name	Delivery Year	Engine	Model Series	Order Month	Order Year	Region	Delivery Total	Order Total	Unfilled Orders
0	Afghanistan	Ariana Afghan Airlines	1968	PW	727	Mar	1968	Central Asia	1	1	NaN
1	Afghanistan	Ariana Afghan Airlines	1970	PW	727	Apr	1969	Central Asia	1	1	NaN
2	Afghanistan	Ariana Afghan Airlines	1979	GE	DC-10	Sep	1978	Central Asia	1	1	NaN
3	Afghanistan	Ariana Afghan Airlines	NaN	CF	737-700	Nov	2005	Central Asia	0	4	NaN
4	Algeria	Air Algerie	1974	PW	727	Jan	1974	Africa	1	1	NaN

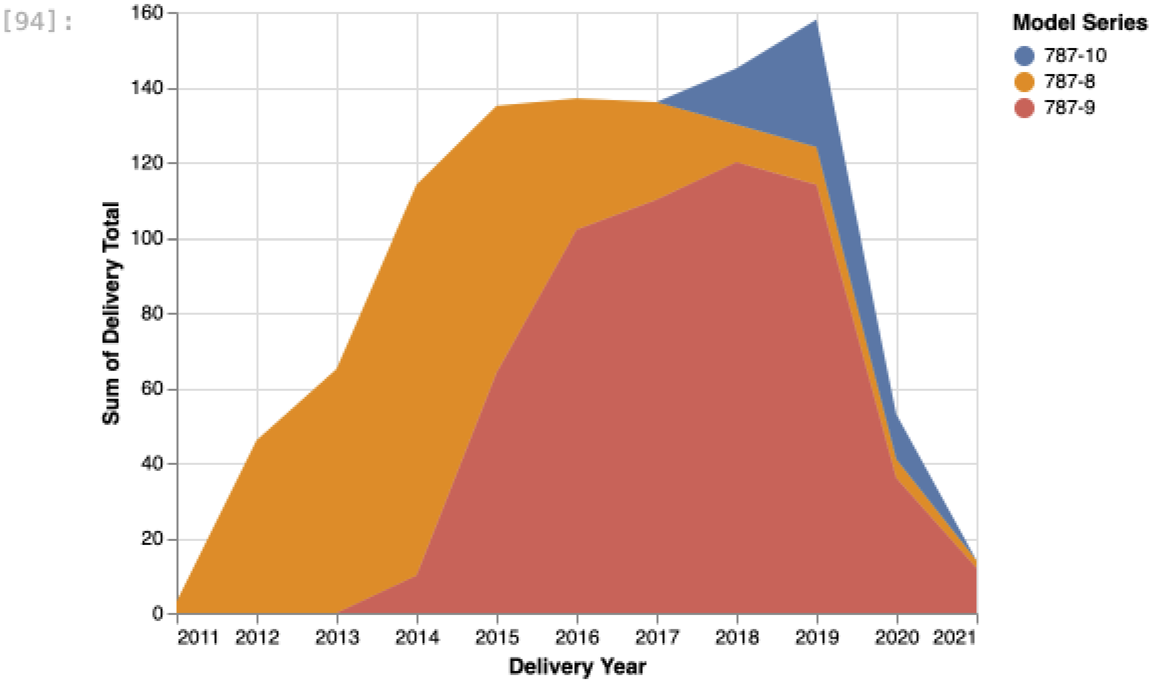
In this analysis, I would like to see the deliveries of certain aircraft models over time. As someone who works in aerospace manufacturing at Boeing, having a visualization of deliveries of specific aircraft models over time shows capability and progress. These types of visualizations are frequently used from the mechanics, to supply chain, and all the way to executive leadership to show how far Boeing is in reaching its delivery goals and contract promises.

Preliminary Visualization

Oftentimes at Boeing, these visualizations are called skylines. These skylines give a bird's eye view past deliveries whether the aircraft delivery was this year or several years ago. Initially, I came up with a stacked area chart. The stacked area chart is good at showing the changes in delivery year by year instantly, It is also good at showing which configurations are being delivered. It is important to know which configuration of each model is delivered since they often use a large number of common parts and come off the same assembly line. Below is a stacked area chart of all 787 deliveries over the years.



```
[94]: alt.Chart(dreamliner_deliveries).mark_area().encode(
          x='Delivery Year ',
          y='sum(Delivery Total)',
          color = 'Model Series'
      )
```



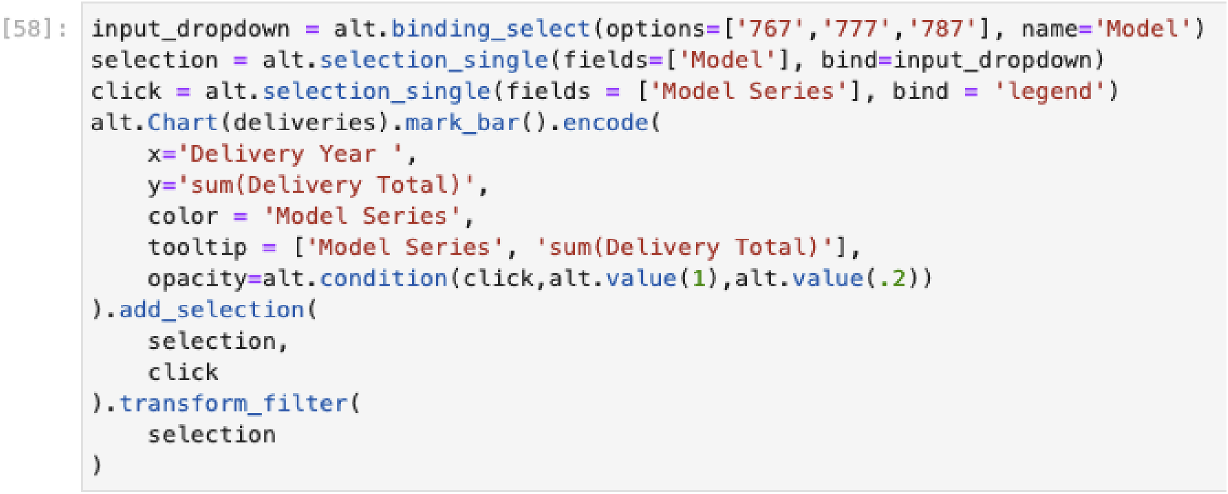
Peer Review

After consulting some of my coworkers, they were able to immediately understand what the visualization was trying to convey. However, after understanding the initial information such as changes in delivery year and what model aircraft contributed to the visual, they immediately started asking other questions. Some of the questions asked were:

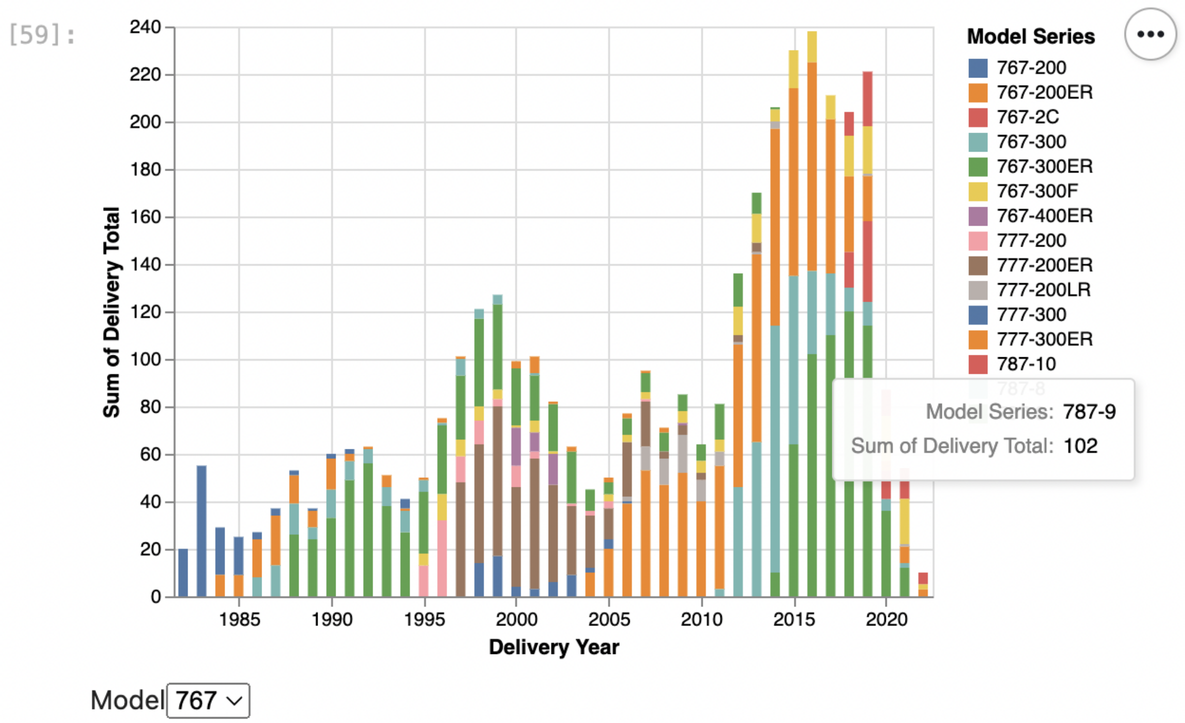
- Exactly how many deliveries of a specific model and configuration were delivered in a year? Am I able to easily access this information?
- The chart is slightly hard to see exactly how many deliveries there were in a specific year at a glance.
- Am I able to see other aircraft models?

Visualization Improvement

After getting feedback for the original visualization, I changed a few things. Firstly, instead of using a stacked area chart, I used a stacked bar chart instead. Using a stacked bar chart gives the user the ability to see the changes in deliveries over time but also has a discrete bar so the user can also easily see a specific year. A screenshot of the chart can be seen below. In addition, I added a tool tip so the user can see which model is selected and how many of that configuration were built in that year.



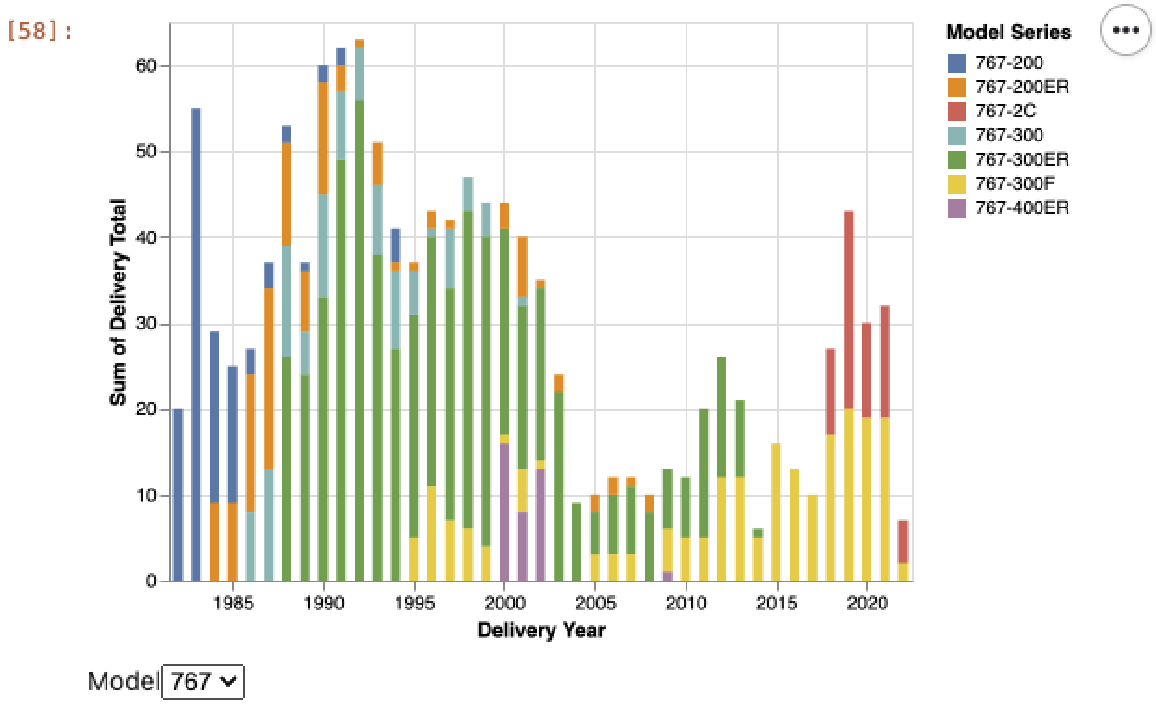
```
[58]: input_dropdown = alt.binding_select(options=['767','777','787'], name='Model')
      selection = alt.selection_single(fields=['Model'], bind=input_dropdown)
      click = alt.selection_single(fields = ['Model Series'], bind = 'legend')
      alt.Chart(deliveries).mark_bar().encode(
          x='Delivery Year ',
          y='sum(Delivery Total)',
          color = 'Model Series',
          tooltip = ['Model Series', 'sum(Delivery Total)'],
          opacity=alt.condition(click,alt.value(1),alt.value(.2))
      ).add_selection(
          selection,
          click
      ).transform_filter(
          selection
      )
```



Model Series: 787-9
Sum of Delivery Total: 102

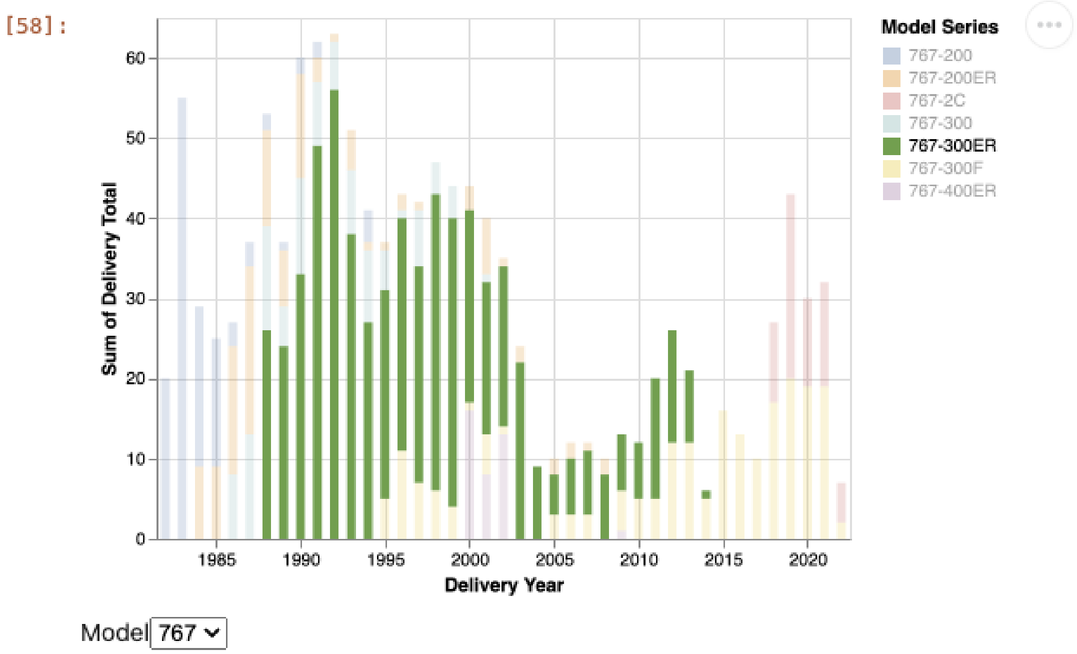
Model 767

Next, I added a dropdown menu so the user can select aircraft deliveries by program. For example, Boeing has several aircraft programs. For this chart, I am able to select between the 767,777, and 787 program. Instead of having to see all of them on one graph or being only able to have one program, the user can now switch between multiple programs. A screenshot of the chart can be seen below. The dropdown menu is at the bottom left and can select which program the user wants to look at.



Model 767

Finally, I was able to highlight the stacked bar chart by the aircraft configuration as well. Being able to highlight all the bars with the same aircraft configuration gives the user the instant ability to see when and how many aircraft are built in that particular configuration over the years. A screenshot of the chart can be seen below.



After these design changes, I showed these visualizations to my coworkers. They all said that the design changes were beneficial and addressed the design flaws of the original visualization. I found that tooltips were useful in specifying any details that the users were looking for after seeing the big picture. Some possible improvements for the future would be to have a secondary chart showing only what a user has selected. Then the user can narrow down to even more specific data.

Notebook Link

Here is the Python notebook link for anyone interested in taking a look at the file. The Python code has already run so there is no need to run it again. You should be able to interact with the charts without running the code.

https://colab.research.google.com/drive/1M_C0-yMf7aFRCmjBeAOW9zYYY_n1NAMm?usp=sharing