Cody Whitt
pkz325
CPSC 4530
Spring 2023
Assignment 4

Github for Project: https://github.com/codydev901/4530_assignment4

Dataset 1: WW2 Allied Bombing By Theater/Aircraft (Tree)

Source and Inspiration

I decided to again use a WW2 bombing dataset I'd found on Kaggle and have used in different ways in a couple of earlier assignments since I find it interesting and adaptable to different questions/visualization methods. This time I decided to look at a comparison of both the magnitude of bombs dropped between theaters, and the aircraft that dropped them within each theater since I felt it would be a good use-case for a treemap.

Link:
https://www.kaggle.com/datasets/usaf/world-war-ii?select=operations.csv

Raw Data Filename: operations.csv

Dataset Type

This dataset is in the form of a flat table. It exists as a single .csv file which was downloaded from the above link and used locally from then on. This one is completely static, since WW2 is over. I suppose it does have the potential to change if errors are found or new records uncovered. Adaptation into a tree structure occurred during the pre-processing step and will be discussed there.

Data Type

Data exists in this dataset in the form of items and attributes. Items exist as individual rows in the table and the attributes are defined as the column labels (the first row of the .csv). The data contains 178281 items and 46 attributes in its raw form, where each item could best be described as an entity holding information representing the occurrence of a mission flown by a particular type of aircraft on some date, somewhere, dropping some amount of bombs in WW2. Certain attribute values shared by the items were used to obtain an eventual tree structure.

Attribute Types and Semantics

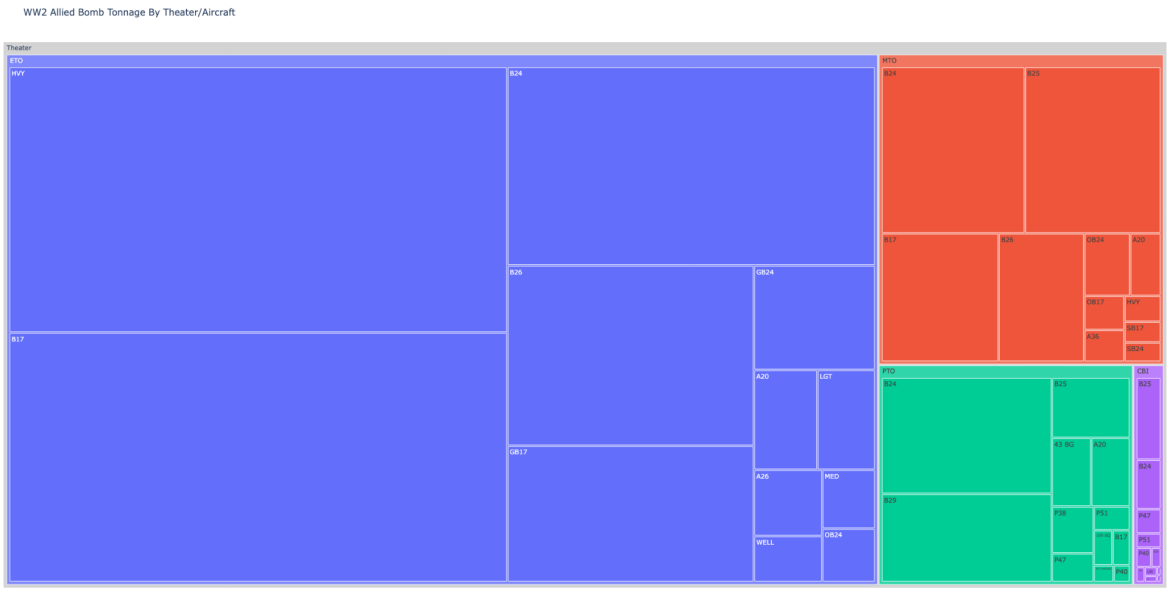
I ended up using 3 of the native attributes for this analysis: *Theater of Operations*, *Aircraft Series*, and *High Explosives Weight (Tons)*. *Theater of Operations* associates each item (mission flown) to what are commonly known as the different major “theaters” of WW2 (Pacific, European, Mediterranean, China-Burma-India). *Aircraft Series* is a categorical attribute holding a string that states which type of aircraft performed the mission, and *High Explosive Weight (Tons)* is a continuous float that states the magnitude of bombs dropped during that mission. *Theater* and *Aircraft Series* served as links between the items.

Pre-Processing and Plotting

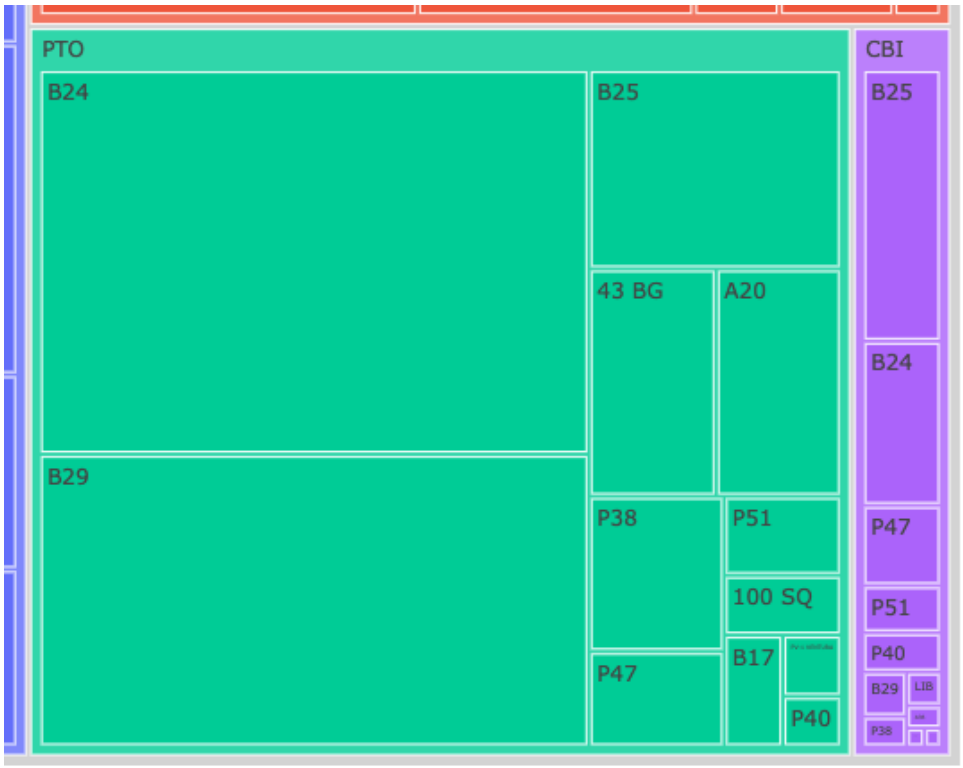
I used Python to process and plot the data, in particular pandas and plotly. After reading in the raw data, the attributes (columns) I wasn’t interested in were dropped as well as null-values. This took the number of items from 178281 to 144107. I then renamed the raw attributes to make them easier to work with and did a quick check of bomb tonnage by theater in order to check the values that would eventually exist at the first level of the tree (the different theaters). I decided to then discard the East Africa theater since its total tonnage was roughly ~1000x less than the next lowest. This reduced the items to 144076. For the final transformation step, I then grouped the items by (theater, aircraft) and took a sum() of the bomb tonnage for each of those pairs. I ended up discarding (theater, aircraft) pairs that made up less than 0.05% of bomb tonnage dropped within their associated theater. This then led to a transformed dataset of 46 items, where each row held a (theater, aircraft, bomb tonnage) item, with theater and aircraft serving as links between the items. The root of the tree is the theater attribute itself/the war overall, then the first level holds the 4 unique theaters, then below that are the aircraft used within each theater (which also have associated bomb tonnage information). Bomb tonnage was then encoded into the rectangle sizes of the plot. To plot, I used px.treemap() provided by PlotlyExpress. This then allowed comparison of the total weight of bombs between theaters, and between different aircraft within each theater.

Parsed Data Filename: ww2_bombing_parsed.csv

Figure



Full Figure



Zoom Screenshot

Analysis

So the objective of this visualization was to allow the viewer to see how overall bombing in terms of tonnage dropped compared between the four theaters of the war as well as to see which aircraft within each theater were responsible for dropping those bombs. At the first level of the tree (one below root), the different *theaters* exist as internal nodes and are color-encoded in addition to their rectangular grouping. At the second level of the tree, the different *aircraft* within each theater, which are leaf nodes, are encoded by text label and rectangular grouping. *Bomb tonnage* at each level of the tree is encoded by rectangular size.

By looking at the size differences of the colored rectangles, it is able to be seen that the majority of bombing activity by tonnage occurred in the European Theater, which is perhaps expected. One thing that was surprising to me is that the Mediterranean Theater (the red group) actually had more bomb tonnage than the Pacific Theater (the green group). When one thinks of WW2 from an American perspective, the European and Pacific theaters come to mind, with the Mediterranean being more of an afterthought, so seeing that more bombs were dropped there than in the Pacific is interesting. When looking at the aircraft themselves, things mostly line-up with expectations. Commonly known bombers such as the B-17, B-24, and B-29 are shown as being responsible for the majority of tonnage dropped. It is worth noting that fighter aircraft such as the P-47 and P-51 are able to be seen in both the Pacific and China-Burma-India theaters which shows that their relative contribution of fighters to the bombing effort in those theaters was more than in Europe and the Mediterranean.

Dataset 2: Delta Airlines Connections and Flight Delay (Network)

Source and Inspiration

I wasn't quite sure what dataset I was going to use for this part, but had a general idea of what to look for in the effort of being able to repurpose one (some sort of categorical relationship between items along with some supporting data that would be interesting to encode on the edges). I found a dataset on Kaggle with information on flight delays for commercial airlines and saw that it would be a good fit.

Link: https://www.kaggle.com/datasets/usdot/flight-delays?select=flights.csv

Raw Data Filename: flights.csv, airports.csv, airlines.csv

Dataset Type

This dataset is in the form of a flat table. It exists as a single .csv file which was downloaded from the above link and used locally from then on. This one is completely static, since it holds information for only 2015. Adaptation into a network structure occurred during the pre-processing step and will be discussed there.

Data Type

Data exists in this dataset in the form of items and attributes. Items exist as individual rows in the table and the attributes are defined as the column labels (the first row of the .csv). The data contains 5,819,079 items and 31 attributes in its raw form, where each item could best be described as an entity holding information representing the occurrence of a commercial airline flight associated with a major US airline in the year 2015. The original source for the data per the Kaggle page is the DOT. Certain attribute values shared by the items were used to obtain an eventual network structure.

Attribute Types and Semantics

Attributes of interest to this project were *AIRLINE, ORIGIN_AIRPORT, DESTINATION_AIRPORT, DEPARTURE_DELAY,* and *ARRIVAL_DELAY*. The first three held categorical strings, while the two delay attributes held both negative and positive integers. The *AIRLINE* attribute associated each item with a major US airline (such as Delta, American etc), while the *ORIGIN* and *DESTINATION _AIRPORT* attributes told where that flight took off and landed in terms of IATA code. The two delay attributes gave information about whether or not the flight occurred on time, which is of particular interest when dealing with air transit. By using the airport IATA code as nodes, each item provides a directed edge from *ORIGIN_AIRPORT -> DESTINATION_AIRPORT* which allows for the dataset to be used in network-type visualization using a node-link layout.

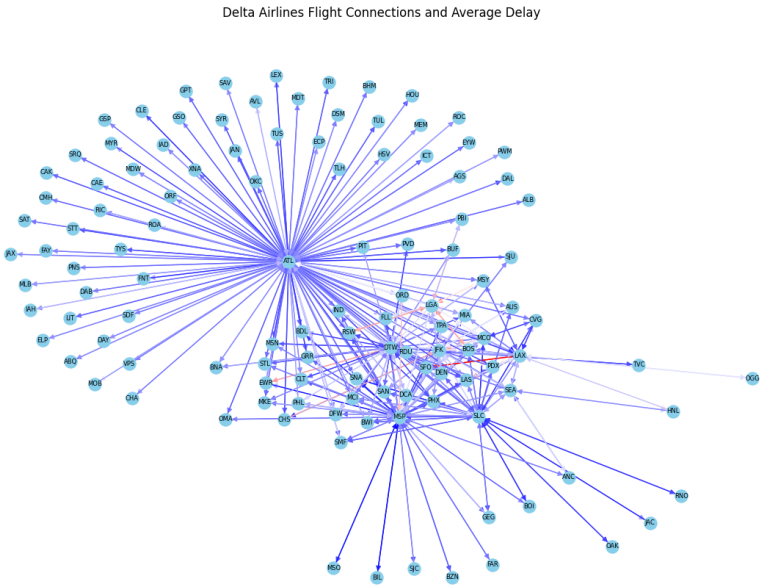
Pre-Processing and Plotting

Python was used to process and parse the data for this dataset, particularly pandas, matplotlib, and networkx. After reading in the raw data, I first filtered to only include flights from Delta. I chose to do an initial data reduction here on *AIRLINE* because I felt the visualization would best be done in context of a single major carrier. After this I then dropped the attributes I didn't plan on using along with null values. This took the number of flights from ~5.8 million down to ~870k. For the final visualization, I was interested in three pieces of information: *origin_airport, destination_airport*, and *average_delay*, where each (origin, destination) represented an edge that could be encoded with an additional piece of information (average delay). To calculate *average_delay*, I grouped the flights by (*ORIGIN_AIRPORT, DESTINATION_AIRPORT)* and took an average of both the grouping's *DEPARTURE_DELAY* and *ARRIVAL_DELAY* separately, then averaged those two values to arrive at a single value representing an average total delay associated with each edge. After an initial visualization, I then came back to this step and chose to exclude edges that had less than 365 flights (roughly once a day service) in order to clean things up/remove flights that were perhaps uncommon. Some sanity checking showed airport codes that didn't appear to be IATA codes in the data, so one final step then occurred where I cross-referenced against provided IATA codes (airports.csv) and removed the questionable values. I didn't end up using it, but I also recorded the total number of flights existing at each edge. This data was then written to a new .csv to be used in the plot set and comprised 496 items each holding *origin_airport, destination_airport*, and *average_delay*.

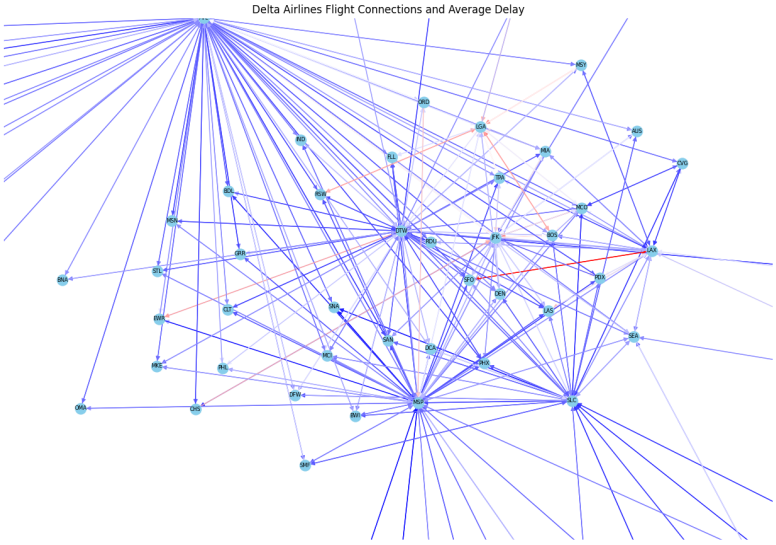
To plot, networkx was used to generate a force-directed graph using the transformed data arrived at per above. The airport IATA codes exist as nodes, and the *origin_airport -> destination_airport* provides a directed edge. To add some additional information to the figure, the edge's were color-encoded with *average_delay* in order to see where Delta experienced the most delay in its 2015 flights.

Parsed Data Filename: flights_parsed.csv

Figure



Full Figure



Zoomed Sub-Region

Analysis

I chose Delta because it's the airline I usually fly on and knew its main hub was in Atlanta, so I was curious (if expecting) the figure to show that. It may be somewhat hard to see, but the airport codes are present within the nodes, and that node in roughly the middle/left with the highest degree is ATL. Someone who wasn't familiar with Delta could infer that ATL is the carrier's most important airport in terms of providing service. It's not really new information, but just visually seeing how all the regional airports connect to a single hub is kinda cool. I didn't look at other airlines, but it would be interesting to look at the rest of them to compare the degrees of their N highest nodes etc.

For encoding *average_delay*, it was mapped from blue -> red, with red having higher delay. One interesting thing here is that those edges (red edges) didn't involve the ATL node, suggesting that where Delta did have higher average delays, it wasn't involving its main/most important airport. The highest delays existed between SFO and LAX, though I can't really think of a potential reason. Some wikipedia searching mentions LAX is the world's 6th busiest airport, so perhaps it would have a high potential for delay, but ATL is the world's busiest overall, which in turn nullifies that argument. Both directions of the SFO / LAX pair were the two highest, which perhaps makes sense in that arrival delays could in turn affect departing delays, especially if the same aircraft is turning around and running the same route back.