Cody Whitt
pkz325
CPSC 4530
Spring 2023
Assignment 1

Github for Project: https://github.com/codydev901/4530_assignment_1

Dataset 1: Russian Personnel Losses in the Ukraine War

Source and Inspiration

This dataset is one I found on Kaggle while browsing through the various available datasets. I have an interest in military history and since this is an ongoing event that is highly publicized I thought it would be interesting to try to do something with it. After looking at a preview of the data I decided I wanted to use it in a way that examines a relationship between day of the week and personnel losses.

Link:
https://www.kaggle.com/datasets/piterfm/2022-ukraine-russian-war?select=russia_losses_personnel.csv

Raw Data Filename: russia_losses_personnel.csv

Dataset Type

This dataset is in the form of a table, in particular a flat table. It exists as a single .csv file which was downloaded from the above link and used locally from then on. In terms of this project it is being used statically, but it appears to be updated weekly and given that this is an ongoing event, has the potential to be used in a dynamic fashion, such as on a website that pulls this file and updates a visualization whenever the dataset is updated.

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 346 items and 5 attributes in its raw form, where each item could best be described as an entity holding information associated with a single day between 02-25-2022 and 02-05-2023.

Attribute Types and Semantics

As mentioned above, the data contains 5 attributes. Of interest to this project were *date* which contained a date string in the format of %Y-%m-%d, and *personnel* which contained an integer representing the number of personnel losses that occurred up to that date. While the

items are technically independent at the raw level, I needed to calculate the daily personnel loss which required using them in pairs during preprocessing. Of the other 3 attributes not used, one simply represented an incrementing count indicating “days since the war began”, and the other two were vaguely related to prisoners of war. These were not relevant to the question I was interested in asking so I ignored them. With *date* being the temporal attribute, the *date/personnel* pairs in each item exhibit time-series semantics as it is an ordered sequence of time-value pairs.

Pre-Processing and Plotting

I used Python to process and plot the data, in particular pandas and matplotlib for this dataset/figure. The data was read directly from .csv into a pandas dataframe and checked for null values (there were none). The columns (attributes) that were not of interest were then dropped. Since the *personnel* attribute held troop losses in a cumulative/up to that day fashion, I then used the built in .diff() function to create a new attribute I called *daily_loss* and assigned it to each item. This takes the value of *personnel* at record i and subtracts the value of *personnel* at record i-1. So for example, if there were 4500 cumulative losses on 2/27/22, and 4300 cumulative losses on 2/26/22, 200 losses are then associated with 2/27/22. This made the assumption that losses represented by *personnel* represent the total lost by the end of that particular day. Date strings from *date* were then converted to DateTime objects and the day of the week determined for each item by accessing an attribute (dayofweek) on the DateTime object and indexing it against a lookup I created to return user-friendly values such as “Monday”, “Tuesday” etc. This was then stored on each item as *day_of_week*.

I was then interested in determining what I described as “the most dangerous day” during each week of the conflict. I decided to simply compare the *daily_loss* at each *day_of_week* during each week of the conflict, where a week ran from Monday -> Sunday (inclusive) and record which day had the most losses for that week. Since the data began on 02-25-22 which was a Friday, I decided to discard the first three data points since that week was incomplete. I also felt that the first days of the invasion would potentially not be representative of the behavior of the conflict as a whole so there is an additional argument there. I got lucky and the data ended on 02-05-23, which was a Sunday, so no items were removed from the end. This resulted in a transformation to a new dataset which contained two attributes *day_of_week* and *weeks_most_dangerous* where the temporal dimension attribute is categorical weekdays and *weeks_most_dangerous* is an integer value attribute that represents how many times that particular week day had the highest personnel loss in a particular week. This new dataset was then written to a file and plotted as a bar chart using matplotlib, where the horizontal axis shows the *day_of_week*, and the vertical axis *weeks_most_dangerous*.

Parsed Filename: rus_dd_week_parsed.csv

The parse and plot steps for this DataSet exist as separate functions and are contained in the parse_plot_russia.py file.

Figure

Analysis

I wasn't quite sure what to expect but it does appear some trends/behavior may be observed from the final figure.

First it appears that Tuesday and Friday could be considered "the most dangerous" days of the week to be a Russian soldier in that these days had the highest loss rates throughout the 48 weeks examined.

Second, days of lower loss rates followed the days of higher loss rates when looking at the relationship between Tuesday/Wednesday and Friday/Saturday which seems reasonable as time would be needed to regroup/consolidate after suffering losses.

Third, weekend losses are lower than weekday losses which suggests combat is less intense on weekends. People typically have weekends off in the civilian world, so perhaps some of that behavior is also present in the military.

Finally, in terms of relating the results to unrelated temporal behavior, Tuesday and Friday also both seem to be associated with "doing things" in the civilian world in terms of things like product/news releases, so perhaps the same holds true for combat operations in Ukraine as well.

Dataset 2: Crime Frequency in Chattanooga

Source and Inspiration

For my final project in CPSC 4180 last semester I looked at criminal trends between the years of 2018 and 2022 in the context of potential relationships with the onset of the COVID-19 pandemic in 2020 for the cities of Boston, Chicago, and Memphis. That project was on a monthly and yearly interval, so I decided to look at crime again here, except for Chattanooga this time and on a weekly/hourly level. I downloaded this dataset from what appears to be Chattanooga's data portal by exporting it in .csv format. The question I wanted to look at was to examine relationships between the frequency of occurrence of particular crime types and day of week/hour of day.

Link:
https://www.chattadata.org/Public-Safety/Police-Incident-Data/jvkg-79ss

Raw Data Filename: Police_Incident_Data.csv

Dataset Type

Similar to the first one, 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. In terms of this project it is being used statically, but again has potential to be used dynamically.

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 507186 items and 19 attributes in its raw form, where each item could best be described as an entity holding information representing an individual occurrence of a crime. Unlike the first dataset, the temporal attribute could not be considered a unique key since multiple crimes may occur on the same date.

Attribute Types and Semantics

The attributes used for this project were *Date_Incident* which represented the temporal attribute and contained a date string in the format %Y-%m-%d %H-%M-%S %p and *Incident_Description* which held and string and contained a simple description/classification of the type of crime that the item represents. The raw data contains a number of spatial attributes, but I did not use them for this analysis. The source of data itself does lend a spatial dimension in that it represents crimes in and around the city of Chattanooga.

Pre-Processing and Plotting

The raw data .csv was first read into a pandas Dataframe and examined to see which attributes were of interest. These were found to be *Date_Incident* and *Incident_Description*, with details of their associated values mentioned previously. All other attributes were then dropped and our two fields of interest checked for null values. Roughly 500 of ~500k values contained a null field and were then dropped. Since each row (item) represents an occurrence of a crime, I wanted to then check how many unique *types* of crimes were being tracked and how many crimes of each type there were. This came out to 101 different types of crimes, with values ranging from ~105k for “Miscellaneous Information” to 1 for “COVID19 Related”. Since many of the crime types could be considered obscure, I decided to limit the scope to 10 types that I felt were self-descriptive and are commonly associated with the concept of crime. I sorted by number of occurance and then manually picked the first 10 candidates I felt were reasonable, these ended up being: Simple Assault, Destruction/Damage/Vandalism of Property, Shoplifting, Theft From Motor Vehicle, Drug/Narcotic Violations, Burglary/Breaking and Entering, Aggravated Assault, Motor Vehicle Theft, Driving Under The Influence, and Robbery.

The data was then filtered to drop *Incident_Descriptions* not in the above list, which brought us down to around ~136k items. I then converted the date strings into DateObjects and through those assigned two new attributes: *day_of_week* and *time_of_day*, were *day_of_week* represents the name of the day (Monday, Tuesday etc) and *time_of_day* mapped to one of four categories: early_morning, late_morning, afternoon, and evening using the hour of the DateObject. I chose 12am-6am to represent early_morning, 6am-12pm to represent late_morning, 12pm-6pm to represent afternoon, and 6pm-12am to represent evening (end-exclusive). I also discarded another ~1.4k items at this step for noticing they all occurred at exactly 1 second after midnight, which I considered suspect/perhaps not representing when they actually occurred that day. This left us looking at ~135k instances of crime belonging to one of 10 types with an associated day of week and time of day that occurred between the years of 2015 and 2023.

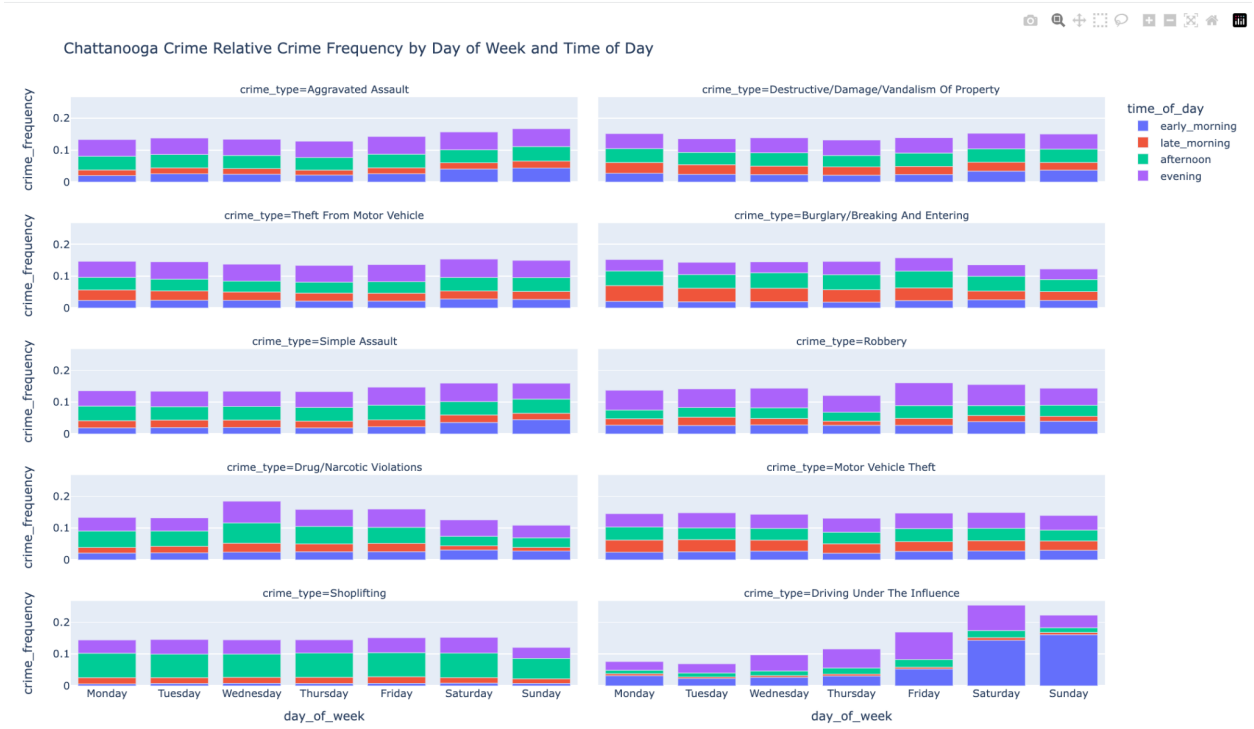
To normalize the data and make final preparations for plotting, I created a new table with the following attributes: *crime_type*, *day_of_week*, *time_of_day*, and *crime_frequency* which was populated using the ~135k remaining items arrived at above. Items were grouped by *Incident_Description* (*crime_type*) where the count represents the total number of crimes that occurred of that type. That group was then filtered by (*day_of_week, time_of_day)* pairs to obtain the number of crimes of a given type that occurred on a given day of the week at a certain time of day. This number was then divided by the total number of that crime to obtain a relative percent I termed *crime_frequency*. This resulted in a final dataset of 280 items, where 10 crime types * 7 days of week * 4 times of day = 280, each having a crime frequency relative to the crime type. This data was then written to a .csv and used in the plot step.

To plot, I used plotly this time and again used a bar chart as the base. Day of week was placed on the x-axis and crime frequency on the y. Time of day was expressed using the color channel, where the stacked bar shows both intra-day behavior through color, and inter-day behavior through height. Facetting was used at the crime_type level to provide separation/create subplots.

Parsed Filename: chat_crime_parsed.csv

The parse and plot steps for this DataSet exist as separate functions and are contained in the parse_plot_crime.py file.

Figure



Analysis

So first addressing some things that I would consider to be expected/help validate the preprocessing. When looking at the Driving Under The Influence subplot, we see higher bars for the weekend compared to weekdays, this fits with the common expectation that DUI's are expected to be higher on those days. We also see that they occur mainly in the evening and early morning hours, which again makes sense. Shoplifting being lower on Sundays perhaps makes sense because stores are more likely to be closed. Breaking and Entering being slightly lower on the weekend could be explained by people more likely to be at home than at work.

In terms of things that I found unexpected/interesting, a couple things that stand out are Drug Violations being highest on Wednesdays and Robbery and Motor Vehicle Theft decreasing on Thursdays, both of which I can't really think of a good explanation for. I would have expected Drug Violations to be highest on the weekend, but they are in fact the lowest then. Perhaps since the police are busy with the elevation in Assault/DUI on weekends, they have less time to deal with Drug Violations.

Dataset 3: Allied Aircraft Usage in WW2

Source and Inspiration

Similar to the first data set, this one was also found on Kaggle and it sparked my interest due to a military history connection. I wasn't exactly sure what I was going to do with it at first, but saw it contained dates and plane types, so I decided I'd start parsing and then decide what questions the data could lend itself to. The questions I ended up looking at/trying to address with the resulting figure are as follows: 1. Which planes were used in each Theater of WW2?, 2. If a plane was used in that Theater, which years was it used and how did its use that year compare to other years it was used? 3. How did the use of a single plane type in a given Theater/Year compare to other planes also used that Theater/Year?

Link:
https://www.kaggle.com/datasets/usaf/world-war-ii

Raw Data Filename: operations.csv

Dataset Type

Like the first two, 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 long over. I suppose it does have the potential to change if errors are found or new records uncovered.

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, in WW2. I ended up using 3 of the native attributes for this analysis, *Mission Date* which provided the temporal information, *Aircraft Series* which provided an unordered classification, and *Theater of Operations*, which provided a spatial classification. The values for these attributes exist as strings in the raw dataset. There are many additional attributes of various types/meaning, so this dataset has the potential to be used in many ways.

Attribute Types and Semantics

Per above, the attributes I was interested in were *Mission Date*, *Aircraft Series*, and *Theater of Operations*. *Mission Date* stores a date string in the format %m/%d/%y and represents a time when a mission took place. *Aircraft Series* held a string containing which gives a brief, but unique reference to what one would consider a type or model of aircraft, for example "B17" or "P51". There were 72 unique types of aircraft in the dataset, I didn't recognize all of them. There is also potential for some of the types to be referring to the same thing which I will address later on. *Theater of Operations* contained 6 unique values. I ended up keeping 4

and discarding 2 which I will discuss below. These were also strings, and the 4 kept were: “CBI”, “ETO”, “MTO”, “PTO”, which I believe refer to the China-Burma-India, Europe, Mediterranean, and Pacific theaters.

Pre-Processing and Plotting

The raw data .csv was first loaded into a pandas Dataframe and basic checks run to see the column types and potential null values. I found that the attributes I was interested in were mostly non-null, with the highest occurrence occurring in *Theater of Operations* where ~3k/~178k were null. After dropping the attributes I wasn’t interested in and then dropping rows that had null data, I was left with ~175k items. I then ran some checks to look at the uniqueness of the categorical attribute values and some quick analysis to look like what I would describe as a distribution of aircraft use (group by *Aircraft Series*, count, and divide against total items). I saw that only 23 of the aircraft types (*Aircraft Series*) were associated with > 1% use. I considered dropping these since it would reduce complexity, but decided not to and went ahead with the processing since I felt there would be value in seeing that relationship (large number of plane types, but many not used that much in relation to others). I did drop two *Theater of Operations* values here: East Africa and Madagascar, since they combined only accounted for only 155/175k total missions (items). Not mentioned above, but each item also contained a *Mission ID* attribute and I ran a quick check to see if it was unique and it was. This suggested there was no relationship between items and the next processing steps would be reasonable to pursue. Additionally a transformation was applied to the *Mission Date* values where I extracted the year and discarded the daily/monthly information, since I was interested in changes between years.

Using the filtered data frame arrived at above, I then set up a new data frame where I would calculate and store the results of the normalization step/use for plotting. This contained the following attributes: *year*, *theater*, *aircraft*, *abs_mission_count*, *relative_mission_count_theater_year*, *relative_mission_count_aircraft_year*. Year, Theater, and Aircraft are the same as the above 3 attributes discussed. *abs_mission_count* holds an absolute count of missions associated with the (year, theater, aircraft) grouping when applied to the filtered dataframe. The other two attributes I’m not that happy with the names, but they represent the following information:

1. *relative_mission_count_theater_year: abs_mission_count* / sum() of mission counts that occurred for all aircraft in that (theater, year) – Attempts to measure how often a type of aircraft was used in a (theater, year) compared to all other aircraft used in that (theater/year)
2. *relative_mission_count_aircraft_year: abs_mission_count / sum()* of mission counts that occurred for that particular aircraft across all years in a theater. — Attempts to measure how the use of a single aircraft type varied across the years within a theater.

I calculated these values iteratively by filtering first in *theater*, then using that to obtain the count of total missions run in that (theater), I then looped through *aircraft* to get count of total missions for (theater, aircraft) and again through *year* to obtain (theater, aircraft, year). The appropriate divisions then occurred and the new data frame populated. Not all aircraft were present in each theater/year. The final data frame was reduced to 311 items with 5 attributes of interest to

display. This data frame was then sorted by *aircraft* and written to .csv to be used in the plot() step.

I used the Python package plotly again here to produce a figure. It was a challenge due to the number of unique categorical values existing for the *aircraft* attribute in particular and could very likely be done a better way, but I felt it allowed for the answer of the three questions poised earlier in a reasonably quick way, which I'll discuss in the Analysis section. I ended up using a group of polar plots, with 4 subplots representing the 4 *theaters*. The radial axis encoded the temporal data *year*. Theta was assigned to the *aircraft* type. The presence or absence of a point reflects the use of an aircraft that year from a binary perspective, then the points are colored to map *relative_mission_count_aircraft_year*, and vary in size to encode *relative_mission_count_theater_year*. An annotation is placed at the bottom of the figure to better explain the color/size channels to the user. The radial axis scale of 1938-1946 was done on purpose to try to make things a little cleaner, though it is potentially misleading. No data points exist for those years.

Parsed Filename: ww2_aircraft_missions_parsed.csv

Figure (Full)

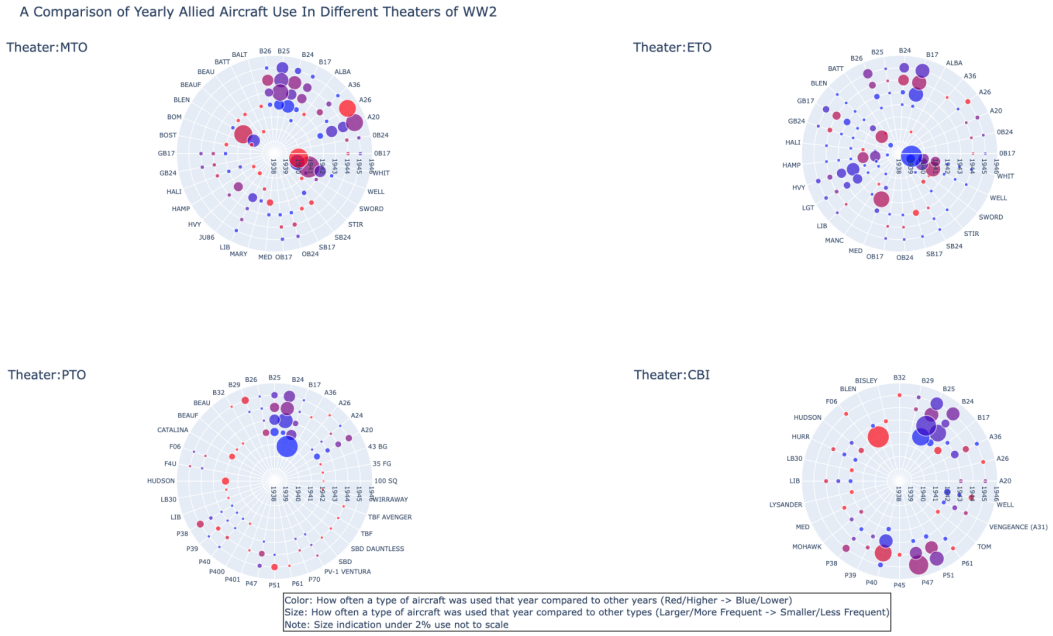


Figure (Zoom 1)

Theater:PTO

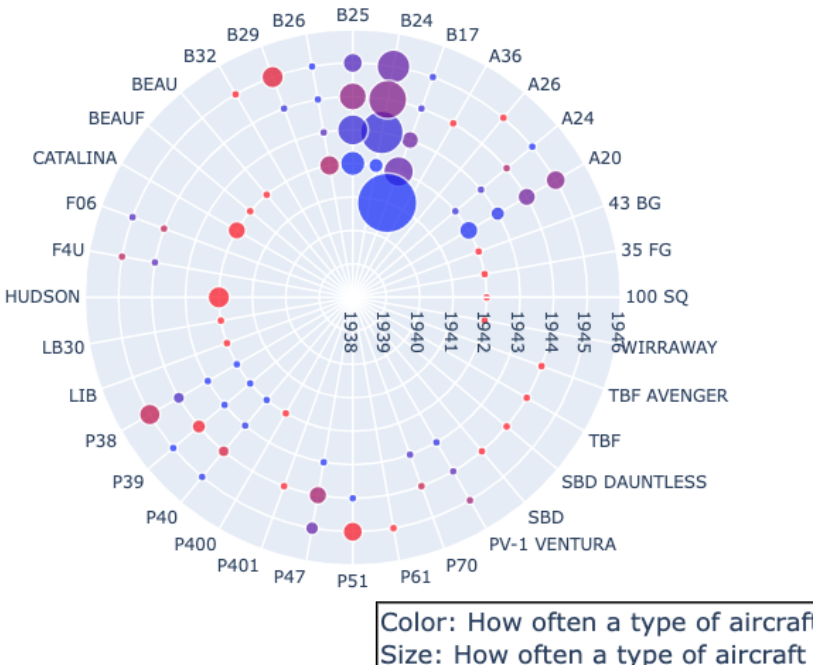
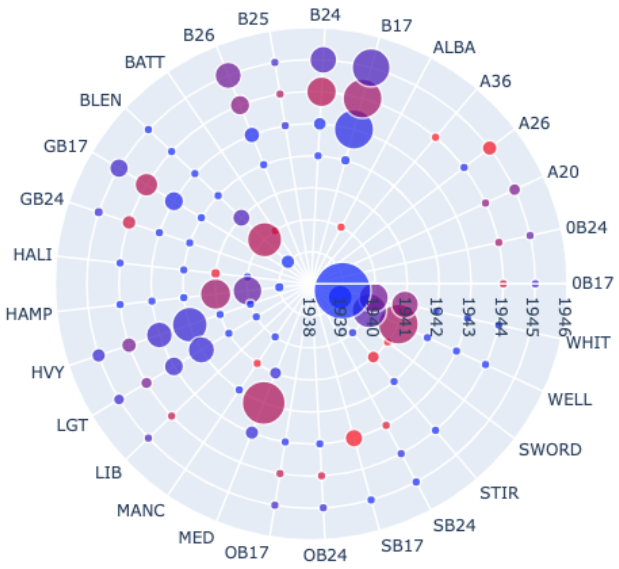


Figure (Zoom 2)

Theater:ETO



Analysis

Note: I included a couple images zoomed in on particular regions since it appeared to be difficult to discern at this lower pixel count.

So using the figure to attempt to answer the questions poised at the beginning, the first would be comparing which aircraft are used in each theater. This can be seen by the presence (or absence) of the aircraft's type at a theta position. These are sorted alphabetically to make it slightly easier. A real world validation case here would be the presence of B29 in PTO and its absence in ETO. This makes sense since B29's were not used in Europe.

Next to look at if an aircraft was used, what years was it used? This can be seen by the presence (or absence) of points along the radial axis. The B29 aircraft again gives a good reference here as it was introduced later in the war, and only has points for 1944 and 1945. Looking at its color channel, we see it is red for 1945, and blue for 1944, which reflects it being used more in 1945 than 1944.

Finally when using the size channel to compare use between different aircraft in the same year, we can see how certain aircraft such as the B17 and B24 were used more often in relation to others which seems to fit the historical perception of those aircraft being well-known.

Having some background knowledge in this area, one thing mentioned earlier that stands out to me would be the possibility of duplicate aircraft types. I didn't attempt to address these since perhaps they have meaning, but for example (GB17, 0B17 and B17) in the ETO sub figure could potentially be grouped as just B17. Another thing that was nice to see is the greater size of points representing British aircraft in the early years of the war (such as WELL - Wellington and BLEN - Blenheim) which makes sense since the United States had not entered the conflict. Unlike the first two figures/analysis, this one doesn't really attempt to answer an unknown question, but to just display the underlying data in a way that would allow for the user to casually determine a quick/rough answer to those specific questions, which I felt were questions one might be interested in in relation to the dataset.