Felipe Castellanos and Charlie Ratliff Final Project

Dataset

We used the dataset "Airports, Airlines, Planes, and Routes". This is a comprehensive dataset that contains over 10,000 datapoints compiled by OpenFlights. The dataset contains air travel data from the year of 2024. The dataset contains four different CSV files to store this data.

1. Airlines.csv:
   This file contains information on every airline company in the world in 2024.
   Items: Airline Name, IATA Code, Country of Origin, Airline ID, Callsign, Active
   Attributes: American Airlines, AA, United States, 24, American, Y

   Note: This file was not used in our final project.

2. Airplanes.csv:
   This file contains information on all of the different airplane models utilized in 2024.
   Items: Airplane Name, IATA code, ICAO Code
   Attributes: Boeing 717, 717, B712

   Note: This file was not used in our final project.

3. Airports.csv:
   This file contains information on all of the airports in the world.
   Items: Name, City, Country, **IATA Code**, ICAO Code, **Latitude**, **Longitude**, Altitude
   Attributes: Waterloo Airport, Waterloo, Canada, **YKF**, CYKF, **43.46**, **-80.38**, 1055

   We used this file to obtain the **geographic locations** of each airport. Specifically, we used the IATA Code, Latitude, and Longitude. IATA Codes were chosen over names or cities since they were present for all airports in the dataset and matched those in the routes.csv. We were able to match the geographic location with the bubble size for each airport using the matching IATA Code in routes.csv.

   Note: Airports without latitude or longitude data (157 out of 14,110) were excluded from our visualizations.

4. Routes.csv:
   This file contains information on every possible flight path in the world.
   Items: **Source Airport**, Source Airport ID, Destination Airport, Destination Airport ID

Attributes: **AER**, 2965, KZN, 2990

We used this file to determine the **number of outgoing routes** from each airport. Using Python, we created a dictionary where keys were IATA Codes of source airports and values were the number of times they appeared in the file. This data was then used to vary the size of the bubbles on our map based on outgoing flight volume.

Cleaning:

The dataset required minimal cleaning, as it was largely pre-processed. All entries in routes.csv and airports.csv included valid IATA Codes. However, 157 of the 14,110 airports (1.11%) were missing either latitude or longitude coordinates. We excluded these entries from our visualization because without geographic coordinates, we could not determine where to place them on the map. Additionally, it is important to note that outgoing flights originating from (but not ending at) these excluded airports were removed during data processing.

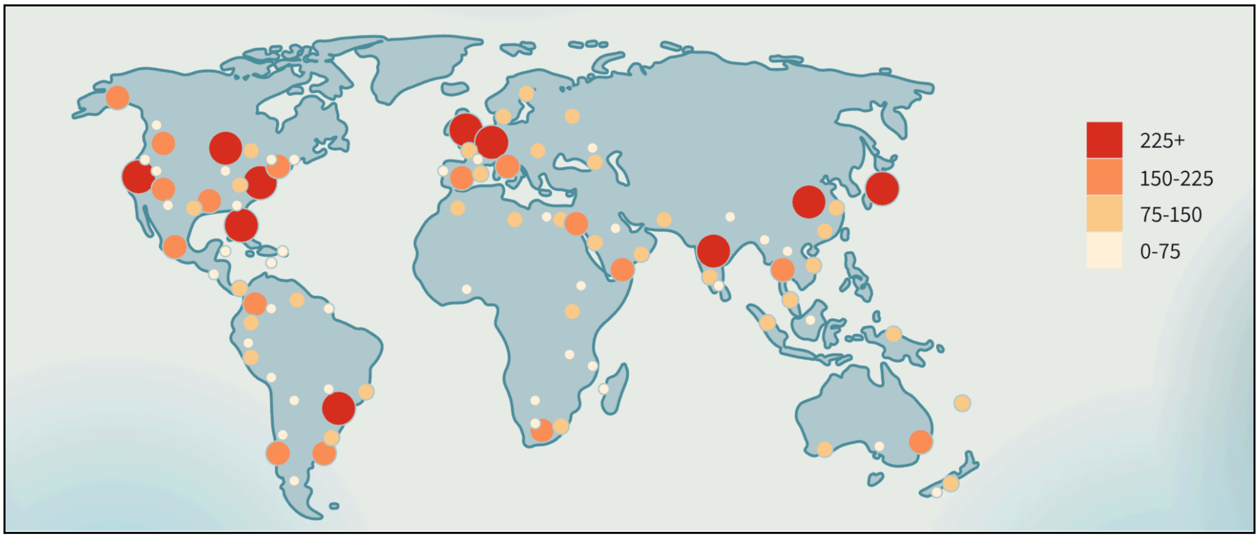
Attributes:

Our visualization relies on several key attributes derived from the dataset. The IATA Code is a **nominal attribute** used to uniquely identify each airport and link the airport data to the route data. Latitude and Longitude are **quantitative attributes** used to determine bubble placement on the Leaflet geographic map.

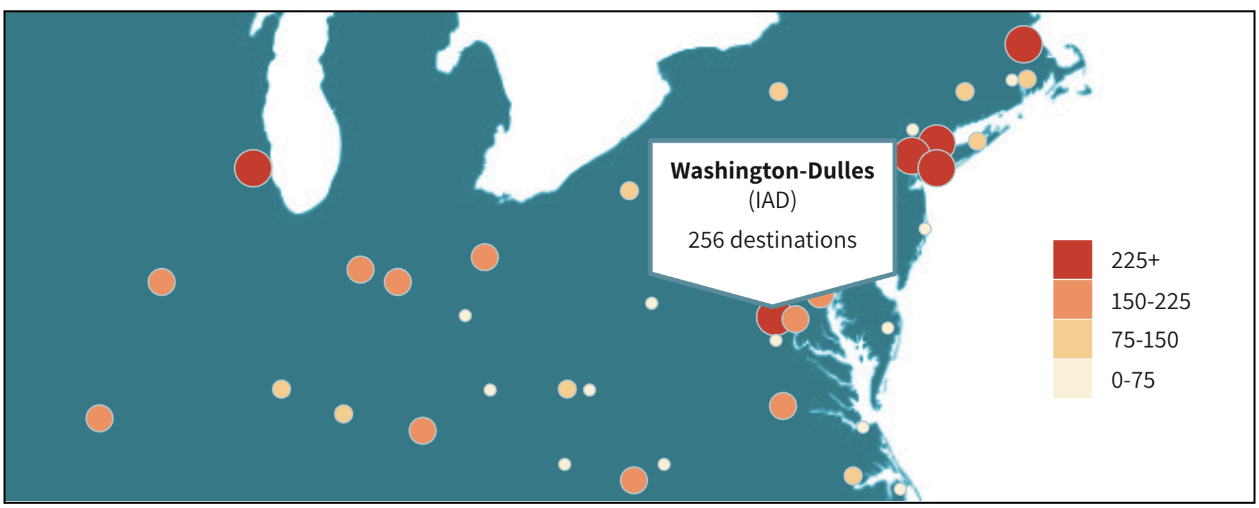
The primary quantitative attribute visualized is **"Destination Count"**, which represents the number of outgoing routes from an airport. We map this variable to both **bubble size** (area channel) and **color hue** (color channel) using a threshold-based color scale. The bubble size is scaled to allow better perceptual understanding of differences across airports with small vs. large route counts.

Additionally, cities and airport names are **categorical (nominal)** data shown in tooltips for clarity but not used in spatial encoding. These attributes assist in comparisons and identifying airports during task completion.

Paper Prototypes



This is the initial visualization presented to the viewer. It displays a world map with bubbles positioned at each airport location. The size of each bubble corresponds to the number of outgoing routes—larger bubbles represent higher amounts. Additionally, a color-blind-friendly gradient color scale is used to indicate the outgoing flight amount, providing another attribute to easily segment different airports' volume.



This is the zoomed-in view of our data visualization. We will implement interactive zoom functionality to allow users to explore specific regions and airports more easily—especially those with a smaller number of outgoing routes. To maintain a clear and usable view, we've set a maximum zoom level of 12.

Our visualization includes a tooltip interaction for enhanced user engagement. When a user clicks on an airport bubble, a tooltip will appear displaying the airport's IATA code and the number of outgoing routes. To meet accessibility standards, we also provide alt text that conveys this information for screen readers. As well as a separate alt text description for the overall view of the world map when not focused on one airport.

Task Analysis and Accessibility Analysis

Our user will be able to accomplish at least three different task using our visualization.

1. Compare**:** *Which airport has more outgoing routes, Charlotte Douglas International Airport (CLT) or Hartsfield-Jackson Atlanta International Airport (ATL)?*

To compare the number of outgoing routes of the two airports and determine which airport has more outgoing routes, the user looks at the color hue, area, and tooltip of the bubble. The outgoing route count is encoded using **circle area** (mark = circle; channel = area) and **color hue** (channel = color). Larger circles and deeper color hues indicate more outgoing routes. The user must zoom into the map to find both CLT and ATL. If the bubble of the two airports is a different color hue or noticeably different area, the user can determine which airport has the larger number of outgoing routes. If not, the user can use the tooltip function to determine the exact number of outgoing routes each airport has and compare this number. The user finds that ATL is encoded in a different color than CLT and uses the key to determine that ATL has more outgoing routes than CLT. If they would like the precise value, they would select ATL and determine it has 915 outgoing routes, then select CLT and determine it has 301 flights.

Accessibility Analysis:

In order to abide by accessibility standards, we assume the user can not use any visual details to complete these tasks. In order to complete the compare task without using a visual aid, the user must use the tooltip function. Since comparing the area and color hue of the bubble are both visual methods, the user must use the tooltip to get the precise value for the airport's outgoing routes. In order to do this, the user will tab through the bubbles on the map, the screenreader will read aloud the information that is in the tooltip when enter is pressed. Once they reach the desired airports, they are able to determine the amount of outgoing routes for each airport using the screenreader on the tooltip.

2. Range: *What is the range of the number of outgoing routes per airport in the state of Nevada?*

Each airport is represented by a **circle mark**, with **area encoding the number of outgoing routes**. Users can compare the **largest and smallest circles** to estimate range, or use tooltips for precise values. In order to determine the range of the number of outgoing routes per airport in Nevada, the user looks at the area/color hue of each bubble in Nevada. They then use this information to determine the largest airport and smallest airport by outgoing flight number (are able to test multiple airports if it is not clear which airport is the biggest or smallest). The user can then use the tooltip function to determine the exact number of outgoing routes. The user finds that PIB has 1 outgoing route and JAN has 18 outgoing routes. Doing the math, the user concludes that the range is from 1 to 18, or (18 - 1) 17.

Accessibility Analysis:
In order to abide by accessibility standards, the user will again use the tab feature to navigate between the different airports in Mississippi. Since the user can not compare the area/color hue of the bubbles, they must iterate through each airport bubble (7 in Mississippi). The user can then use the information on the tooltip to read the number of outgoing routes, keeping track of the largest and smallest values that they find. After going through all 7 airports, they are able to find the range to be 1 to 18, or 17.

3. Order: *Order the top 5 busiest airports in the world by number of outgoing routes.*

To order the top 5 busiest airports in the world by number of outgoing routes, the user will need to first use color hue/area. Users can infer order by using the filter dropdown menu and identifying the **largest circles** (mark = circle; channel = area); they then confirm values using **tooltip text**, which provides precise outgoing route counts. **Color hue** also reinforces differences in volume. The area will be crucial in this, as the top five busiest airports by number of outgoing routes will have much larger bubbles by area than the other airports. Once the user has determined a set of airports that are the busiest five (with the addition of more to ensure they got the five busiest), the user is able to use the tooltip feature to see the exact number of outgoing routes for each of these airports. The user can then order these numbers from largest(busiest) to smallest(5th busiest). The user would find the largest bubbles by area are ATL(915), ORD(558), CDG(524), LHR(527), and PEK(535). The user then orders these to find the result of ATL, ORD, PEK, LHR, CDG.

Accessibility Analysis:
In order to accomplish the order task while following accessibility standards, the user must tab through each airport bubble. The user can keep track of the largest airports(by number of outgoing routes) using the tooltip and screen reader features. Once they have traversed the map, the user can then return the correct top five busiest airports in the world.

Pilot the Tasks

User Testing Procedure

To evaluate the accessibility, clarity, and usability of our digital airport visualization, we conducted a structured user testing session with a peer. This tests followed the user study protocol outlined in the final project description, with consistent task wording and a pre-written testing script. Felipe acted as the **Speaker** and Charlie as the **Note-taker** during the session.

The user interacted with the visualization on a **MacBook Pro**, using a trackpad. The session lasted approximately 5 minutes, including task completion and feedback discussion.

Testing Script (Used Verbatim)

Speaker:
"We are evaluating our visualization and are asking you, the participant, to complete some tasks using the visualization and then provide feedback about the visualization and experience. As a reminder, we are evaluating the visualization, not you as a participant, so you don't need to worry about being 'right' as you complete these tasks. There are three tasks, followed by a brief feedback session. The whole pilot session should take under 5 minutes. Do you consent to participate?"

[User responds]

"Thank you for agreeing to participate. We will start with the three tasks. Please 'think aloud' as you complete the task, meaning voice what you are thinking as you work through the task. Your first task is:

1. Compare the number of outgoing flights in Charlotte vs. Atlanta."

[Pause for task completion]

"Wonderful. Your second task is:
2. What is the range of outgoing flights across all airports in Nevada?"

[Pause for task completion]

"Alright. Your third task is:
3. Identify the five busiest airports in the world by number of outgoing flights."

[Pause for task completion]

"That is the end of the third task. For this last bit, we welcome any feedback you may have about the visualization or about your process for completing the tasks."

User Testing Results & Observations

Task 1: Compare Charlotte vs. Atlanta

- The user used the zoom buttons rather than gestures, indicating discomfort with trackpad-based navigation.
- Upon locating two large bubbles in the southeast U.S., the user hovered for tooltip info.
- Confused by the IATA codes, saying: *"I think Atlanta is ATL, I'm not sure what Charlotte is."*
- Successfully identified the correct cities but noted it took long.

Task 2: Range of Nevada Airports

- Scrolled west and zoomed into Nevada region.
- Pointed out three distinct bubbles but paused while the map reloaded: *"It's lagging when I scroll."*
- Found the min and max values by reading tooltips and calculating manually.

Task 3: Top 5 Airports

- Scanned visually across the map.
- Tried to estimate based on bubble size and color.
- Did not notice the filter dropdown until the task was complete.
- Attempted to filter by clicking on the legend bubbles.

Feedback	**Planned Design Changes**
• Wanted full airport names on hover (tooltips were updated based on this). • Wished the legend was interactive as opposed to a dropdown menu. • Suggested adding onboarding instructions as a possible intro modal.	• Tooltips changed to include full name, IATA code, and city. • Make bubbles on legend clickable and dynamic. • Add inline "How to use this map" modal upon start and for persistent guidance.

WAVE Evaluation & Accessibility Fixes

We evaluated our final webpage using the WAVE accessibility tool .

Initial Findings:

- One missing form label associated with our *<select>* dropdown used for filtering airports.
- Absence of semantic page regions, such as *<main>* or *<header>*, triggering a "No page regions" warning.

Solutions Implemented:

- We added a proper *<label for="min-connections">* element to associate the form control with screen reader-friendly text.
- We restructured the HTML to include semantic *<header>* and *<main>* tags, allowing assistive technologies to better understand page layout.

These updates significantly improved screen reader navigation and resolved all flagged issues. No contrast, alt text, or heading structure problems were reported.

Finalized Visualization

https://charlierat.github.io/airport-connectivy-visualization/

User Testing

Participant 1
Non-Davidson Student, Biostatistician, 40s

Device: MacBook Pro (trackpad navigation)

Task 1: Compare Charlotte vs. Atlanta

- Scanned southeastern U.S. and used tooltip to verify ATL and CLT.
- Completed task correctly in under 30 seconds.
- Appreciated tooltip information but said: *"It would help if you could search by city."*

Task 2: Range of Nevada Airports

- Correctly identified airports in Reno, Las Vegas, and Elko.
- Again, mentioned map not loading bubbles immediately.
- Found range by subtracting smallest bubble from the largest.

Task 3: Top 5 Airports

- Didn't notice dropdown filter.
- Guessed visually based on largest visible bubbles.
- Clicked on one, confused by lack of sorting mechanism.

Feedback	**Planned Design Changes**

- Wanted the filtering interface to be clearer and consistent in terminology: *“What are connections? Is that the same as destinations?”* - Skimmed the onboarding modal too quickly: *“Maybe put the tips in the corner?”*	- Replace terms “connections” and “outgoing flights” for consistent terminology. - Add a persistent question mark (“?”) icon for guidance. - Explore implementing search functionality for airport names. - Find a more reliable option to update bubbles when zooming and panning.

Participant 2

Non-Davidson Student, Industrial Engineer, 30s
Device: MacBook Pro (trackpad navigation)
Task 1: Compare Charlotte vs. Atlanta

- Took over a minute to complete.
- Eventually guessed based on general U.S. geography, however had no negative comments on the map itself.

Task 2: Range of Nevada Airports

- Completed without trouble, exactly as intended using the hover feature.

Task 3: Top 5 Airports

- Missed filter completely.
- Sorted visually by guessing largest bubbles globally by comparing them to the legend.

Feedback	**Planned Design Changes**
- Expressed interest in a way to “list or sort” airports before being indicated that that was available. - Liked the colors and map style but said interactions weren’t obvious. - Suggested adding a legend explanation directly in the interface.	- Move dropdown menu to legend box to make it more evident. - Reinforce legend interactivity with tooltips and hover styling.

Participant 3

Davidson Student, Economics Major, 20s
Device: MacBook Pro (external mouse setup)

- **Task 1: Compare Charlotte vs. Atlanta**
  Scrolled to southeastern U.S., but mistakenly compared Raleigh instead of Charlotte.
- After realizing the mistake, correctly hovered over ATL and CLT to compare outgoing flights.
- Noted "the hovering is intuitive".

Task 2: Range of Nevada Airports

- Used hover tooltips to find and write down the highest and lowest values.
- Had trouble locating Las Vegas due to lack of concentration on state borders and zoom level.
- Eventually succeeded by zooming in on Nevada.

Task 3: Top 5 Airports

- Didn't notice dropdown initially.
- Selected "500+ connections," successfully.
- Completed the task as expected, by hovering over the 5 visible bubbles to view the city names.

Feedback	**Planned Design Changes**
• No negative feedback or ideas for improvement • Enjoyed the ease of use	• Implement previously mentioned static "help button" to bring back intro modal.

Personal Reflection

Felipe

This project was the first time we applied the material we covered in class to build something interactive from start to finish. It was really satisfying to see how all the concepts—like visual encoding, interactivity, and accessibility—came together in a real application. A friend who helped with user testing mentioned how cool it was that we could actually build something like this ourselves, it was quite awesome.

At first, I thought it would be more difficult to pull off. But using examples and D3 tutorials helped a lot, and once we had a structure in place, things moved along. There were definitely some challenges, though, especially with bugs and layout issues. A couple times, none of the bubbles would render at all. We had to troubleshoot a lot, but each fix helped us understand the tools better.

One of the biggest takeaways for me was how important user testing is. We thought we were basically done with the project, but once we did the pilot test, we realized there were several things we could improve. Even small changes—like how we worded labels or where things were placed—made a big difference in how people used the visualization. It reminded me that the way users actually interact with something doesn't always match what we expect.

The final round of testing, with three different users, reinforced that. A participant in his 40s was more used to Windows than macOS, and even that affected how he interacted with the map. It was a good reminder that we can't assume every user will navigate things the same way. I'm glad we built something that worked for a variety of users and also met accessibility guidelines.

In the end, I think the process of building, testing, and adjusting taught me as much as the technical side of the project. It showed me how important it is to keep things flexible and to be open to feedback.

<u>Charlie</u>
I really enjoyed working on this final project. I thought that it was really fun to be able to make our own visualization and that we had the tools to create something like this. Something that I really enjoyed about the process was the creativity aspect. I found it fun to be able to go through different design choices and make small changes that would help the user accomplish the desired tasks. I liked how we got to make the project our own and design it however we wanted to.

One thing that I thought was difficult about the project was starting it. I felt like it took us a while in order to come up with what we really wanted to make and what it should display. Even then, we still did not cover everything that we ended up having to do. I thought that making the initial code when we were starting with an empty file was hard. However, the d3 tutorials online helped a lot. Once we were able to get our initial code down, I thought that the project went very smoothly!

I thought that the interview gave insight that was crucial to making our visualization contain everything that we needed. Everyone will view the visualization uniquely, and many of the things we implemented in our first sketch were not inherently clear to a user who has not been working on it for weeks, like we had. So, these tests gave us good knowledge of what features we needed to change, make clearer, or add.

I really enjoyed working with Felipe on this project. I thought that creating our own creative data visualization was very fun! This is something that I hope to be able to do more of in the future, and I am very glad that I have these skills as a developer now.