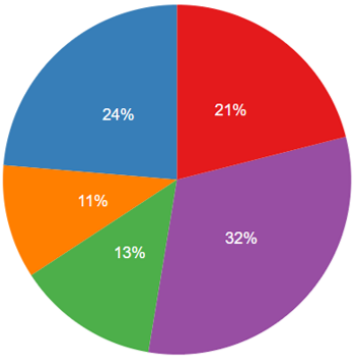
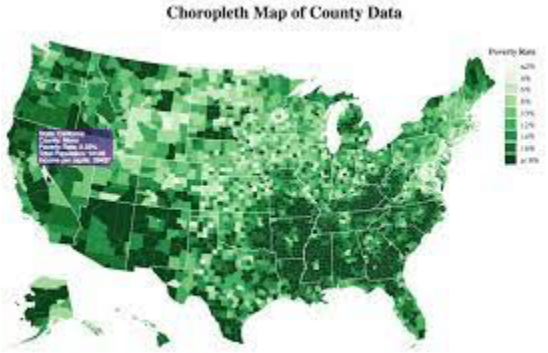
To Do for Next Sprint

Feature 1: Severity Pie Chart



Feature 2: Number of accidents per county



Test Cases

Incremental Analytic 1 Test Case: As a developer, I want to have the severity pie chart to use the incremental analytic approach.

- **Test Case 1:** As a developer, if a user clicks on the pie chart page, I want to store an array of 50 state objects, where each object's value is the number of accidents in that state.
- **Correct output:** The array is created with 50 state objects and stored.
- **Test Case 2:** As a developer, if a user adds a record with the state of CA, I want to increment the appropriate severity object in the array by 1.
- **Correct output:** The array's CA object's value is incremented by 1.
  Test Case 3: As a developer, if a user deleted a record with the state of UT, I want to decrement the appropriate severity object in the array by 1.
- **Correct output:** The array's UT object's value is decremented by 1.

Incremental Analytic 2 Test Case: As a developer, I want to have the state map analytic use the incremental analytic approach.

- **Test Case 1:** As a developer, if a user clicks on the states map page, I want to store an array of 50 state objects, where each object's value is the number of accidents in that state.
- **Correct output:** The array is created with 50 state objects and stored.
- **Test Case 2:** As a developer, if a user adds a record with the state of CA, I want to increment the appropriate state object in the array by 1.
- **Correct output:** The array's CA object's value is incremented by 1.
  Test Case 3: As a developer, if a user deleted a record with the state of UT, I want to decrement the appropriate state object in the array by 1.
- **Correct output:** The array's UT object's value is decremented by 1.

To Do for Next Sprint

Front-end:

1. Feature 1 render as incremental analytic
2. Feature 2 render as incremental analytic
3. Add pages to query page
4. Fix nav-bar
5. Use boot-strap to improve css
6. Fit barchart feature onto page

Back-end:

1. Load csv file with county geolocation & # accidents for county map (Estela)
2. Array for States Map (Incremental Analytic)
3. Array for Severity Pie Chart (Incremental Analytic)
4. index for (county, state) for county map
5. Paginate query page to reduce render time.
6. Implement function to return data for swarm plot

Done List of Last Sprint

Front-end:

- County map render (Estela)
- Feature 5 render Pie chart (Yiu Ming Wong)
- Fix up CSS bar graph(Yiu Ming Wong)
- Fix css calendar heatmap (Ivan)
- Clean up the form css (Ivan)
- Feature 6 render(Jacob)

Back-end:

- A function that returns the start time of accidents for a specific date(Jacob)
- A function that returns the number of accidents per severity. (Yiu Ming Wong)
- Load data from CSV or backup file on server start (Thuan)
- Backup data on server close (Thuan & Ivan)
- Parse the whole CSV file with all the accidents (Thuan)
- Indexing data.json for faster look up(Thuan)
- A function that returns accidents per county.
- County map function to return number of accidents for each city (Estela)