Assignment 1 — Siru ZHONG

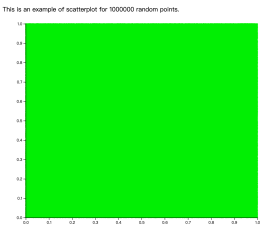
> 💡 *index.html* presents a scatterplot of 10,000 random data points. Try to increase the number of data points to 1 million, 10 million, and even 100 million. Answer the following questions.

1. What problems do you encounter when the number of points increases? (30 points)

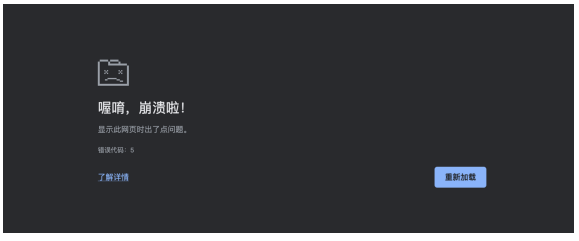
As the number of data points in the visualization increases, several challenges emerge:

1.1 Performance Degradation

- For a dataset of 1 million points, the browser experiences noticeable lag and becomes temporarily unresponsive before finally rendering the scatterplot. This lag is due to the computational overhead of processing and rendering a large number of graphical elements.
- As the dataset grows to 10 million and 100 million points, the performance issues escalate further. The browser becomes severely unresponsive and, after struggling for an extended period, eventually crashes. This is a clear indicator that rendering such vast numbers of points directly is not scalable and leads to a poor user experience.



1 million points



10 and 100 million points

1.2 Visual Overplotting

- The scatterplot for 1 million points, as evidenced by the provided screenshot, showcases a phenomenon known as overplotting. This occurs when data points overlap to such an extent that the visualization becomes cluttered and loses its

informative value. In the screenshot, the dense green coverage makes it nearly impossible to discern individual data points or meaningful patterns. This obscures any insights that might be gleaned from the visualization.

1.3 Memory Overhead

- As the number of points increases, the memory consumption of the browser also escalates. Especially with datasets in the tens to hundreds of millions, browsers might not have sufficient memory resources to handle such a load, leading to crashes or system slowdowns.

1.4 Decreased Interactivity

- Interactive features like zooming, panning, or tooltip displays become sluggish or entirely unresponsive with large datasets. The delay in interaction can hamper the user's analytical workflow and reduce the overall utility of the visualization.

---

2. Which data transformation method would you choose to address the problems? Name two data transformation methods. (30 points)

2.1 Aggregation

This involves grouping multiple data points into a single representative point or bin. For instance, you can aggregate data points that fall within a particular region into a single point or represent them with a heatmap.

2.2 Sampling

This involves randomly selecting a subset of the data to display. This helps in reducing the data size while still representing the overall trend.

---

3. What are the benefits and drawbacks of each transformation method? (40 points)

3.1 Aggregation

- **Benefits**:
    - Reduces the number of points, improving performance.

- Can help in visualizing the density or frequency of points in a region (e.g., through a heatmap).
- Helps reduce overplotting.

- **Drawbacks**:
  - Might lose some finer details or outliers in the data.
  - Requires deciding on the method and criteria for aggregation.

3.2 Sampling

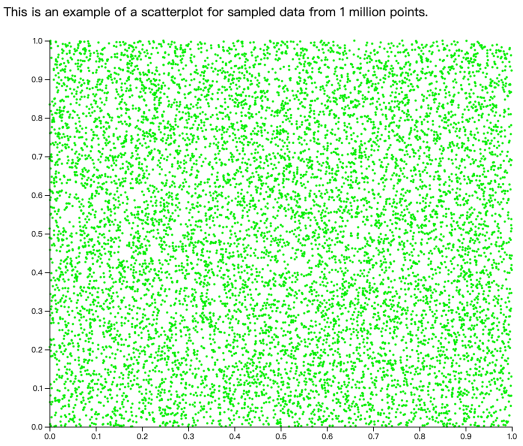
- **Benefits**:
  - Simplifies the dataset while retaining a representative overview.
  - Can significantly improve performance.
- **Drawbacks**:
  - Might miss important patterns or outliers.
  - Random sampling might not always give a representative view of the entire dataset.

---

4. Implement one of the data transformation methods and submit the code. (20 bonus points)

Detailed code can be found in `index_aggregation.html` and `index_sample.html`

4.1 sample



4.2 aggregation