An Insight into Artificial Satellites

Group number G04
Group members
Abhijeet Amle
Bingqian Zhang
CPCS 6030, Fall 2020

Introduction

The intention behind man-made satellites is to get a better understanding of the universe. Satellites are used for many purposes, such as taking pictures of the planets, communication, navigation, safety, and emergency management. They also measure gases in the atmosphere, such as Ozone and Carbon dioxide. For the above purposes, most of the countries are launching their satellites into space. In this project, we are visualizing different trends in the satellite application areas, the number of satellites launched by each country, and how satellites revolve around the Earth in their designated orbits.

Dataset

The dataset is created by the Union of Concerned Scientists (UCS). The dataset is a list of 2666 operational satellites currently in orbit around Earth. Each of these satellites has 28 attributes, which include technical information like mass, power, launch date, the expected lifetime; orbit information such as apogee, perigee, inclination, and period; contains information on who owns, operates, built the satellite, and its purpose.

Dataset link: https://www.ucsusa.org/resources/satellite-database
(The dataset is updated three times a year. For our project we used the April 1st, 2020, release which was the latest version available at the time of the start of this project.)

Design solution

The dataset which we are using has 28 different attributes for a satellite. This dataset is not perfectly cleaned, as we have observed carefully, values for certain attributes are missing. Also, there are a few attributes like the source of the data for each satellite which is irrelevant to the visualization, so we have removed those attributes while exploring the dataset in Tableau.

To design our story about satellites, first, we have started by showing the number of satellites launched by each country. Then we have shown the trends in the application areas of those satellites. After that, we have introduced the user to types of orbits, and in the last visualization, we have shown all the satellites revolving around the Earth in their designated orbits.

Visualization 1: A color map showing number of satellites in all the countries
We want to show all the countries that have launched satellites. We used colorMap instead of heatMap to encode it. There are 241 countries in the world and satellite numbers owned in each country are a little bit polarized, color hue can more clearly encode each country, that's because the difference in color hue is big enough to distinguish encoded categories. But the color map can't encode the satellite number. To differentiate which countries launched the most satellites, we used the color legend to encode the satellite number. ScaleThreshold is applied to color legend and color scale. We have used a tooltip to show the satellite number for each country. The map can be zoomed in and out by moving the mouse over it. With the help of this visualization, we can find out the number of satellites launched by each country.

Visualization 2: Multi-series line chart showing trends in satellites application areas
To visualize trends in the satellite application areas, we used a multi-series line chart. The X-axis represents the year, Y-axis represents the number of satellites, and hue colors encode the category of satellite application areas, marks are lines. Compared to the pie chart and streamgraph, the multi-series line chart, we think is the best one to express the satellite development trend. The line chart encoded satellite development changed from 1988 to 2020, we can see the turning port of each application area from it. Hovering over the line, it gets highlighted to get a move clear picture.

Visualization 3: A 3-D visualization showing different types of satellite orbits around the Earth
For our 3rd and 4th visualization, we want to visualize different orbits around the Earth. To visualize the Earth and the revolving satellites in the designated orbits, we used a 3D visualization with the help of the three.js library. We can visualize the rotation of the Earth and the revolution of all the satellites around it. We can control the speed of each object individually such as Earth is rotating at a constant speed, satellites with different orbits are revolving with respect to their length of the orbit. Also, we have used the light function in the three.js library to lighten one side of the Earth to show the difference between day and night.

Visualization 4: A 3-D visualization showing all the satellites revolving around the Earth in their designated orbit
The final visualization is all satellites revolving around the Earth in their designated orbit. We have used different color lines to represent different orbits and a cube to represent the satellite in that orbit. We have provided a filter option to the user where the user can select a specific orbit to visualize. Also, a user can zoom in and out, also can change the camera angle of the visualization to get different views of the satellite trajectories.

With the help of the above visualizations, users can answer questions such as:

- How many satellites does a given country have in orbit?
- How many satellites are used for military purposes versus commercial purposes?
- When was the oldest working satellite launched?
- At what altitudes do most satellites orbit?

Related work

Currently, there are a few satellite visualization tools available on the internet in which we can configure satellite coordinates and can visualize only one satellite trajectory. Considering the work done in Satellite Data Visualization Application for plotting trajectories of a satellite, they are giving input as all the orbit details and visualizing a satellite trajectory. Also, there is open-source software released by NASA, GMAT, in which one can visualize the trajectory of a single satellite and can have multiple views of it like in a 2D map and a 3D view. Here, http://stuffin.space/, they have visualized all the satellite trajectories in a single 3D view of Earth. There is no filter option available, we can only see the default view of all the satellites revolving around the Earth.

Literature review

We used colormap instead of other graphs for the first visualization because there are too many countries, more than 200. Map is good at showing all the countries very clearly at the same time. And to interpret colormaps, people must determine how dimensions of color (e.g., lightness, hue) map onto quantities of a given measure (e.g., brain activity, correlation magnitude). This process is easier when the encoded mappings in the visualization match people's predictions of how visual features will map onto concepts, their inferred mappings. It is beneficial to use colormaps that will not appear to vary in opacity on any background color.

For our second visualization of trends in satellite application areas, we used the multi-series line chart. A multi-series line chart is basically a line chart with more than one line representing the comparison trends. Here, we are mapping continuous data over a period of time. It is easier for the human mind to understand changes that are happening over a period of time. With the help of a line chart, we can pinpoint the exact time when something out of the ordinary happened. A multi-series line chart helps in comparing different groups on the same scale to find out the key performance areas among those groups. And we can predict and infer each line's development in the future according to the development history of multi-chart.

THREE.js is a JavaScript library designed to simplify 3D rendering in the browser using WebGL. WebGL has a low-level, imperative JavaScript API which makes it relatively difficult to use on its own compared with a higher-level declarative technology like SVG. three.js provides a straightforward and concise API that (while still imperative) makes setting up 3D scenes using WebGL much simpler. WebGL renders scenes to a single <canvas> DOM element that contains the entire visualization on the page, unlike SVG which is composed of a separate DOM element for each element of the visualization.

References

1. K. B. Schloss, C. C. Gramazio, A. T. Silverman, M. L. Parker and A. S. Wang, "Mapping Color to Meaning in Colormap Data Visualizations," in IEEE Transactions on Visualization and Computer Graphics, vol. 25, no. 1, pp. 810-819, Jan. 2019, doi: 10.1109/TVCG.2018.2865147.
   https://ieeexplore.ieee.org/abstract/document/8454346

2. Michael Barry
   http://msbarry.github.io/threejs-tool-page/

3. https://www.fusioncharts.com/line-charts