

I used AI for this project. ChatGPT, Phind, and Claude.

<https://projects.fivethirtyeight.com/congress-trump-score/>

I am visualizing a dataset of senators during the 115th, 116th, and 117th senate during Trump's presidency. It includes basic senator information such as name, party and state.

It also contains data of how often every member of the Senate votes with or against Trump during his presidency. It contains data on

- number of votes cast
- the percent of votes cast aligned with Trump.
- the predicted percent of votes cast aligned with Trump, by the fivethirtyeight statistical model
- the percentage points Trump won (or lost) the Senators state by in the 2016 election.

The purpose of my project was to focus on the percentage of votes cast aligned with Trump.

I chose to use a bar chart as it provides easy comparison of senators in their alignment with Trump. I added additional filtering by party to provide finer grain selection, and to provide the user with the ability to examine differences in the party.

I picked a parallel coordinates chart as I had to represent 5 variables (party, votes cast, predicted %, trump %, and state %) for 100 senators. For so many variables, the parallel coordinates provided the cleanest chart. I also thought the brushing looked like a really fun challenge and wanted to do that.

Also, like in the bar chart, I wanted the viewer to be able to select and examine senators individually, with the parallel coordinates chart providing a great opportunity for this.

I think I did a good job on this project. For the aesthetics, I went minimal and clean. Light theme worked well with tones of blue and red for the parties, which were mandatory. I do worry about the pink selection for the parallel coordinates chart, but I tried a green line and simply darkened the line and preferred the pink.

As for the bar chart, I stuck with the simple style, and I think it paid off. It is not too cluttered, but manages to contain lots of information and 3 sets of independent buttons. With labeled and color coded controls I hope they are easily understood by the user.

As for the rectangles for brushing, I first thought of using the delete key to get rid of rectangles, and a "spawn new rectangle button". Then I realized that I would have to give instructions on deletion, and got rid of that plan. Then I made a trash can, where you could drag rectangles into. It was ok, but it didn't look very nice. Finally, I made a new system, where if you draw the rectangle close to the edge of the screen, it will delete. I thought this was smooth and intuitive. The user can also simply grab as many rectangles as they want from the original, which was pleasantly smooth to implement and use.

*data sourced from Five Thirty Eight*