Jesse W. Fimbres
Aaron J. Heckman
May 8th, 2019

Republicans, Democrats, and Climate Change Opinions in the United States

CS504 Data Mechanics Project Report

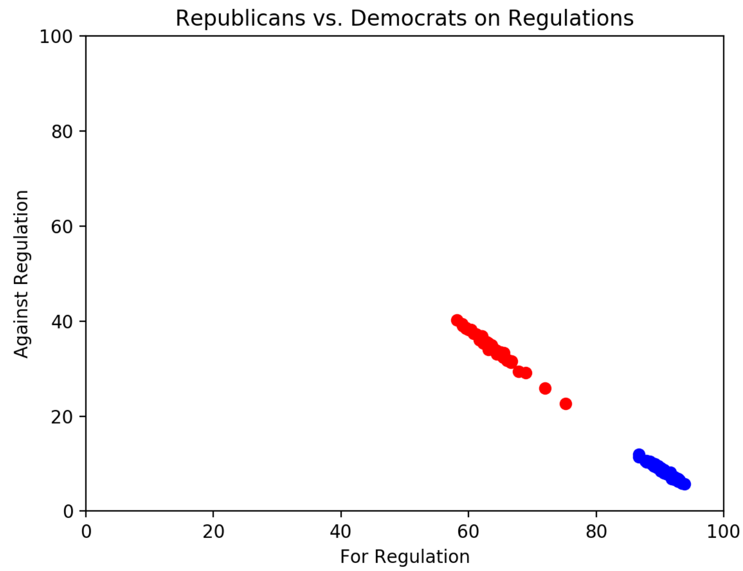
This decade there has been an increasing divide in the United States between its two dominant political parties: republicans and democrats. According to *The New York Times*, **"Partisan identification is now a bigger wedge between Americans than race, gender, religion or level of education"** . This comes at a time many climate scientists view enacting government policies as a necessary solution to prevent irreversible harm to Earth's ecosystems. However, climate change is a very divisive issue between republicans and democrats in a political system only increasing in polarity. This begs the question- is there common ground regarding climate change that can be reached between the two parties? To help answer problems related to this question we thought of what data mechanics tools we have at our disposal. Ultimately, we settled on the following two goals:

1. Determine the degree of difference between republicans and democrats in each state on select opinions relating to climate change to determine which ones **maximize** common ground between the two parties (e.g. Is congress doing enough about climate change?).
2. Find the **correlation** between carbon efficacy and CO2 emissions per capita to determine if an increase in how efficiently carbon is used by a state decreases its emissions per capita and if so by how much.

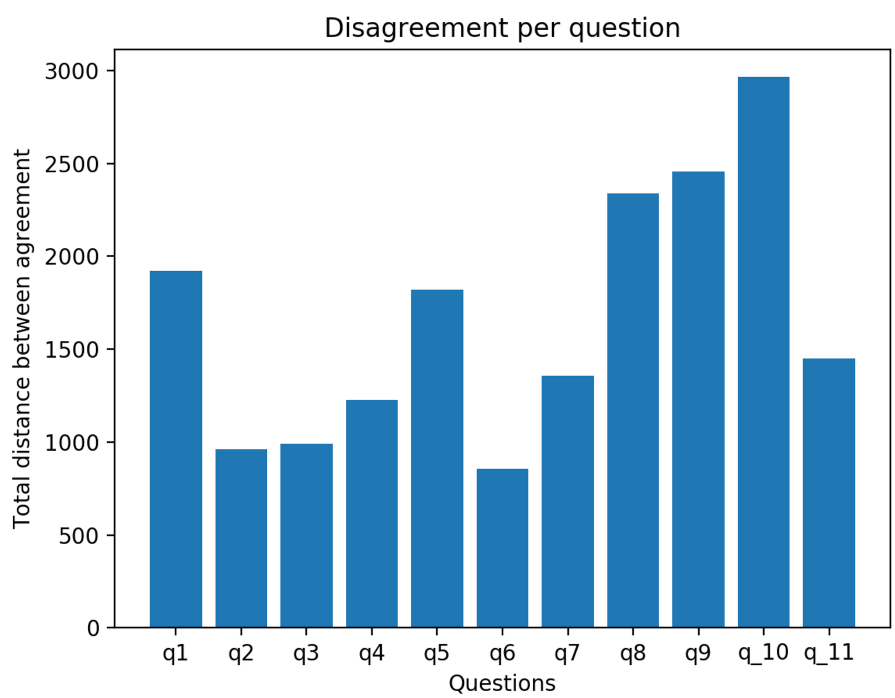
The following datasets were used for the project:

1. United States Partisan Map Data
2. United States Population/State Data
3. State Energy-Related CO2 Emissions Adjusted
4. State Energy-Related CO2 Emissions Unadjusted
5. Carbon Intensity Per State

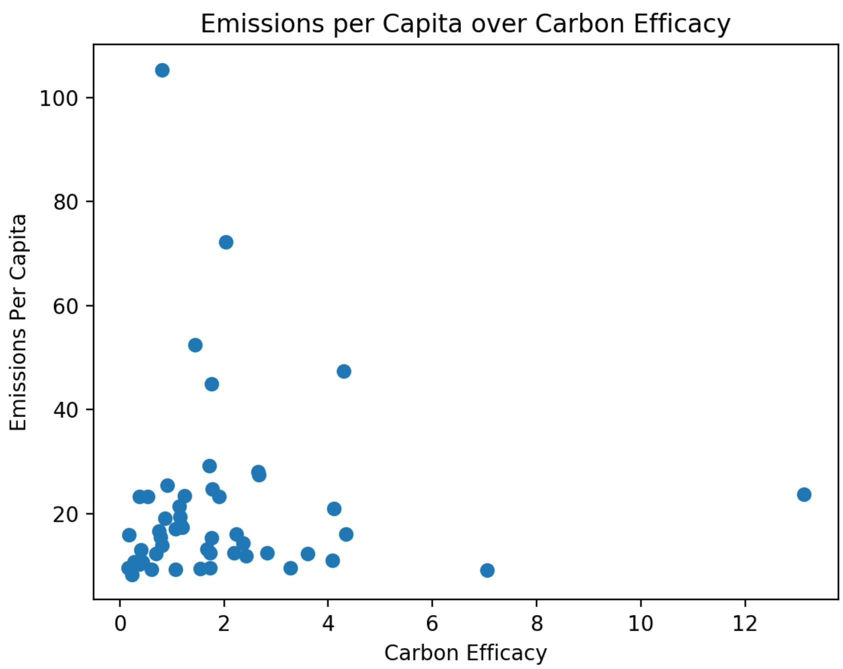
The datasets can be found on GitHub. We use the **get_carbon_data.py** file and the **get_pop_data.py** file in order to pull the information from the datasets of value to the project. Next we create our own modified datasets in **carbon_efficacy.py** and **emissions_per_capita.py** so that we can manipulate the relevant data.



With regards to each opinion: for each state's democrats and republicans, we mapped the party's respective opinion on a scatterplot with the x-axis representing the percentage of people for regulation and the y-axis representing the percentage of people against regulation. Since these values will usually sum to around 100, you'll notice that all the scatterplots follow a linear pattern. Republicans are represented as red dots and democrats blue. On our interactive web component, we have each bar in the bar graph below act as button that returns the respective scatterplot for the opinion asked. The full list of opinions can be found on our poster and on our web project.



Next we use the k-means algorithm to determine the amount of disagreement between the two parties. K-means works by splitting data into a fixed number of clusters (k) and creating data points at the center of each cluster, essentially finding the cluster's average. Since we know republicans and democrats are polarized, with some certainty we are guaranteed at k=2 that one k will represent republicans and the other democrats. From here we find the distance between these two points for each opinion and graph them on the bar graph above in order to visualize the level of agreement between different questions asked. From here we can easily find the questions with the least and most disagreement. This code can be found in the **optimization.py** file on GitHub.



correlation coefficient = -0.06883
p-value = 0.63478
(In layman's terms, there is no statistically significant correlation between emissions per capita and carbon intensity.)

Using our census and carbon emissions data we calculate the emission per capita per state. Next we decided on this formula to represent carbon efficacy: (average of Adjusted and Unadjusted CO2 Emissions) / Carbon Intensity. We then graph the results found for each state on the scatterplot above. Next we use functions from the Python SciPy library to calculate the correlation coefficient and p-value of the data which are displayed above. The **p-value** shows the level of marginal significance within a statistical hypothesis test that represents the probability of the occurrence of a given event. The **correlation coefficient** measures the statistical significance between two variables. Values less than 0.8 and greater than -0.8 are not considered significant. The code for this analysis can be found in the **stat_analysis.py** file on GitHub. On our web component, a form appears below the scatterplot above where users may enter the name of a state (with proper capitalization) and, once submitted, a scatterplot with solely the location of the specified state is returned.

For the web component of the project, we used flask to manage our backend in order to retrieve data from our Mongo database, call the necessary functions to organize and manipulate the data, and connect to our frontend. The specifications to run the web component can be found on GitHub. Our website has two interactable parts, which are briefly mentioned in the methodology descriptions above. In order to accomplish the embedded image buttons for the k-means portion we used HTML image maps and formatted each bar with its respective scatterplot accordingly. The other interactive component for the statistical analysis is simply a form that reads the user input and searches for a state name that matches the respective query. Interestingly enough, while seemingly straightforward, the most time-consuming problem of this project came from this part of the project. We encountered a “ghosting” problem in which flask did not display any updates to the graph made in the backend on the website itself. After many hours of frustration we learned that this was a caching issue, and we had to adjust the cache to store things for 0 hours instead of the default 24.

From the bar graph for disagreement per question we can see that the question with the least amount of disagreement is q6: **It’s worth protecting the environment at the cost of economic growth.** If we were to analyze this data even more, and perhaps given more climate change opinions to work with, we could attempt to find patterns between various opinions and their level of agreement between the two parties. Interestingly we found no relationship between carbon efficacy and carbon emissions per capita. This rejects our initial hypothesis and seems counterintuitive and given more time we would explore the reason for this lack of relationship. If we were to guess- and this is not based in statistical analysis- we think this is because states that use carbon more efficiently also expend more carbon overall, therefore diluting any statistical significance for the relationship. Additionally our unique method for calculating carbon efficacy could be a factor. The sources for this project can be found on GitHub, while the sources used for the report specifically can be found on the web component, with HTML links to the respective sources.