Congressional Vote Clustering Individual Report

Tolulope Olatunbosun (tao5634@g.rit.edu)

For this project I was particularly interested in determining the relationships between each politician's voting record as it related to every other politician. Because our project has a natural breakdown into two particular political parties, I suspected my group would implement clustering techniques. Knowing that some clustering algorithms are vector quantizations that compute distances between data points, I determined that I would find a distance metric that would be helpful in our particular use case. Oftentimes in distance metrics, the Euclidean distance metric is implemented. For example, in the K-Means method, it is based on the Euclidean distance between two data points, because the sum of their squared deviations from the centroid is equal to the sum of the squared Euclidean distances divided by the number of data points altogether. This represents a spatial distance, and the K-Means method is used to minimize squared errors. However, I chose to implement a different distance metric.

I instead implemented the Jaccard Distance metric, which is defined as an intersection of two data sets divided by the union of those data sets. In other words, because I wanted to quantify the similarity between the voting preferences of each politician, the Jaccard distance represented a technique to compare the encoded label vectors for each politician, and compute their voting similarity to others. This would in turn allow me to see whether certain parties had members who colluded in votes, and how different a politician's voting style could be within their party, and compared to a different party. A question that may arise is why did I select the Jaccard distance over the commonly implemented Euclidean distance. In this scenario, I was interested in the distance between the two sets, and although the Euclidean distance is a commonly used method, I wasn't interested in using it because the data set I had wouldn't represent spatial distances. The vectors and data simply represented binary data, which the Jaccard Distance metric is often implemented for. Further, the Jaccard distance is useful for comparing observations between categorical variables, asymmetric binary vectors, and more. Downsides to using the Jaccard distance include the fact that the Jaccard distance metric doesn't quantify sensitivity to sizes of distinct data sets.

To quantify these relationships, a matrix was created to document the similar bill votes between each politician, and this metric was divided by the total number of bills. This matrix provided the information to see who voted similarly and who did not. This information was plotted as the "Democrat and Republican Clusters Based on Jaccard Distance Matrix" graph, documenting the two disparate clusters. This process allowed me to quantify what it means to have different voting preferences and how to represent this through data visualization.

Function for making the Jaccard distance matrix. Finds the distance of one preson to everyone else for all people in the data set.

In [12]:

```
from scipy.spatial.distance import jaccard
def jacMatrix(names, df):
    distances = []
    for i in names:
        person = df[i]
        distance = []
        for j in df.columns:
            distance.append(jaccard(person, df[j]))

        distances.append(distance)
    return distances
```

In [13]:

```
jacMat = jacMatrix(names, data)
```

In [15]:

```
#make a color array corresponding to each person in the data set to represent party
color = []
for i in party:
    if i == "Democrat":
        color.append("blue")
    elif i == "Republican":
        color.append("red")
```

This code uses the distance matrix and networkx to plot all of the congress people based on their distance to everyone else.

In [18]:

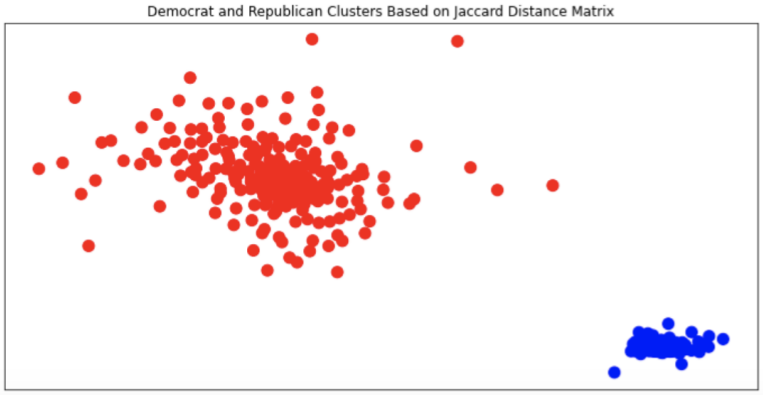
```
dt = [('len', float)]
A = np.array(jacMat)
A = A.view(dt)

G = nx.from_numpy_matrix(A)

plt.figure(figsize = (12,6))
pos = nx.drawing.nx_agraph.graphviz_layout(G, prog='neato')
G.remove_edges_from(list(G.edges()))
G = nx.drawing.nx_pylab.draw_networkx(G,pos=pos,node_size = 100, node_color = color,alpha = 1,with_labels=False)
plt.title("Democrat and Republican Clusters Based on Jaccard Distance Matrix")
```

Out[18]:

```
Text(0.5, 1.0, 'Democrat and Republican Clusters Based on Jaccard Distance Matrix')
```



CONGRESSIONAL VOTE CLUSTERING
Foundations of Data Science and Analytics (DSCI 608)

Le Nguyen (383006341) - ln8378@g.rit.edu
Greeshma Ganji (362007736)(gg1849@rit.edu)
Nandhini Lakshman (nl7222@g.rit.edu)
Tolulope Olatunbosun (tao5634@g.rit.edu)

Problem Statement

In this project, we wanted to see if the current political divide in American politics shows up in voting record data, and furthermore see if we could build a model to classify which party a member of congress belonged to based on their voting record. We suspected that the vast ideological differences between the parties would be very apparent and show up as significant differences in voting records. From this suspicion, we assume the congressional data can be clustered into groups representing each party.

Research & Literature

For this project, we looked at past analyses of this problem. Past work has primarily been done on clustering data from senators, while we clustered data from members of the house (govtrack, 2021)(Swallow, 2017). We can see from historical data, that the senate has been becoming more and more divided over time (Swallow, 2017). We suspect the house of representatives will follow the same pattern and be divided into distinct clusters along party lines. We can use some sort of clustering method to do this.

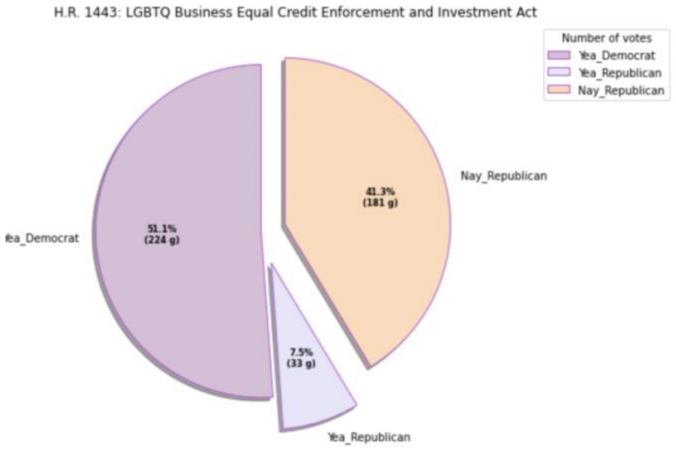
We had to find a proper model to do this and further a distance metric to define distances between congress people to cluster then. We found the Jaccard distance to be the most suitable distance metric to use as it is the distance between two sets (the sets of voting records) (Glen, 2016). Also through our research, we found the most appropriate clustering method to use was hierarchical clustering (Sharma, 2019).

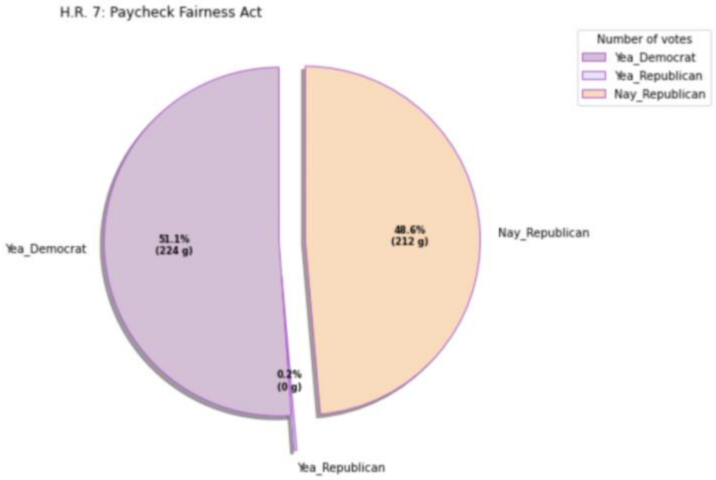
Analysis

Two lists were created for the number of times a Democrat and a Republican voted against their own party. The mean count was computed for each list so we could get a better picture of the average number of outliers. From the results, we observed a higher number of republicans who voted against their party compared to the number of democrats. We wanted to visualize the outliers in the form of pie charts for certain bills.

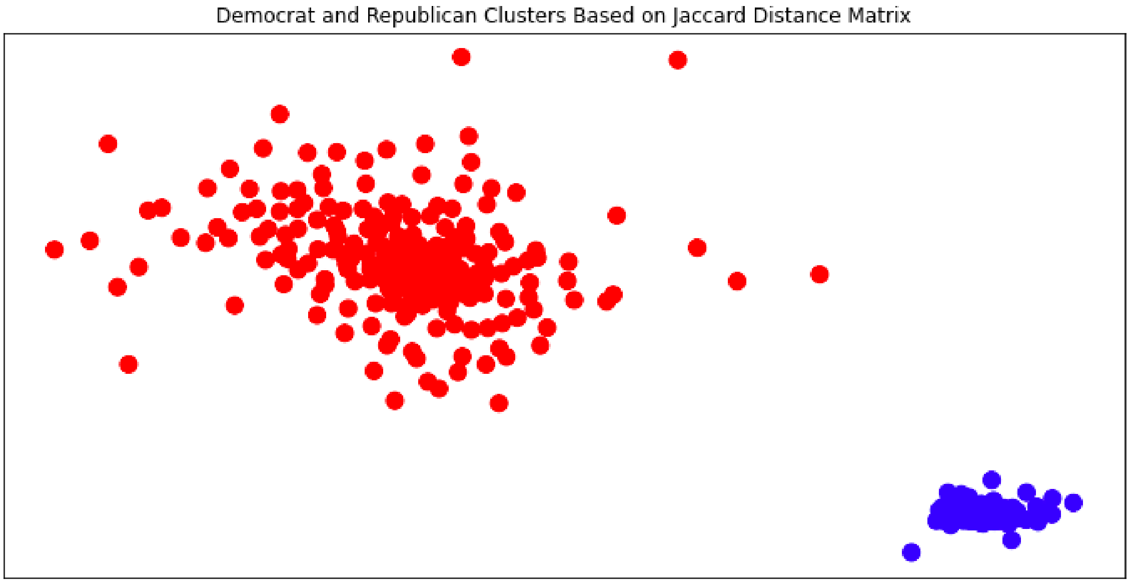
Upon visualization, for certain bills, we noticed a significantly higher number of Republicans deviating from typical Republican voting behaviour. The portions of the pie charts were much smaller for the Democrats who voted against their parties.

Generally, the parties seemed to be very divided in their voting behaviour.





After computing the Jaccard distance, we discovered relational distances between each politician with respect to their voting records. Smaller distances demonstrated similarity between voting choices across various bills as well as similar beliefs, being members of similar parties, or perhaps collusion [Scipy, 2021]. On the other hand, greater distance metrics represent distinct differences in voting preferences. The figure below illustrates each individual senator with respect to other senators. We found that members of the Democratic party generally vote with their party, and often with little variance. Contrarily, Republican senators represented varied voting preferences, though, distinct enough to stay within their respective clusters.



Working Plan

Le - Web scraping and cleaning the data for analysis and model development. Did some data analysis on the side as well.
Nandhini - Performed data visualization and data analysis on controversial bills to inform voting preferences and to detect anomalies in either party.

Tolu - Performed data analysis, and implemented Jaccard distance metric on the data to map preferences between politicians voting behaviors.

Greeshma -

Teamwork & Collaboration

In terms of teamwork and collaboration we did struggle a bit to have consistent meetings because we were all busy with our classes and had conflicting schedules. After the first few meetings to get the project idea set up we did not meet again for about a month. We had to start frequent meetings again when the project due date came near. This was a problem when it came to giving weekly updates because we were not consistently meeting to generate new work and have everyone else informed. Everything did come together in the end though and worked out.

In hindsight having more frequent meetings would have made the project a more comfortable process, but it is understandable that we got caught up in other work and classes. Perhaps having online meetings would have served us better with our busy schedules.

Works Cited

1. govtrack. (2021). *Voting Records*. Retrieved from govtrack.us: https://www.govtrack.us/congress/votes

2. Swallow, E. (2017, November 17). *U.S. Senate More Divided Than Ever Data Shows*. Retrieved from Forbes: https://www.forbes.com/sites/ericaswallow/2013/11/17/senate-voting-relationships-data/?sh=4768ba54031d

3. P. Sharma, "A Beginner's Guide to Hierarchical Clustering and how to Perform it in Python," 27 May 2019. [Online]. Available: https://www.analyticsvidhya.com/blog/2019/05/beginners-guide-hierarchical-clustering/

4. Scikit-learn Machine Learning in Python, Pedregosa *et al.*, JMLR 12, pp. 2825-2830, 2011. https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html

5. pceccon. (2016, March 31). *Generating graph from distance matrix using networkx: inconsistency - Python*. Retrieved from StackOverFlow.com: https://stackoverflow.com/questions/36339865/generating-graph-from-distance-matrix-using-networkx-inconsistency-python

6. S. Glen, "Jaccard Index / Similarity Coefficient," 2 December 2016. [Online]. Available: https://www.statisticshowto.com/jaccard-index/

7. *Scipy.spatial.distance.jaccard*¶. scipy.spatial.distance.jaccard - SciPy v1.7.1 Manual. (n.d.). Retrieved December 2021, from https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.distance.jaccard.html