# Redistricting in Washington State

## Summary:

Through the development of Mixed Integer Programming (MIP) models I am seeking to create a new redistricting plan for the state of Washington. I created 2 different models with varying levels of success by altering which parameters I use to filter data. I mainly wanted to lower the deviation of each district for the state as much as possible while ignoring demographic data to get the truest mix of voter types. Neither of my plans ended up being better than the current district plan however with further refinement a more efficient plan can be calculated.

## Introduction:

Political Districting has commonly been an issue in America since the inception of political districts. One of the biggest issues with districting is creating a plan without allowing gerrymandering (adjustment of districts to suit political gain). This has led to many different ideas about what criteria are important and what model is the best to fit these criteria to create the most balanced districts. This project will go over one method and potential other options that could be used to create a good redistricting plan for the state of Washington. My method mainly seeks to reduce the deviation factor for the criteria. Other popular plans include maintaining compactness and contiguity, focusing on reducing strange shapes through edge factors as well as doing multiple objectives at one time. My objective is one of the simpler mathematical ones, however is important to maintain voting power remains equal.

## Criteria:

According to the National Conference of State Legislatures (NCSL), the traditional districting principles for states involves, compactness, contiguity, preservations of counties and other subdivisions, preservation of communities of interest, preservation of prior districts and avoiding pairing incumbents. These are the generic ones that will apply for every state however Washington state also has prohibited intentionally favoring or disfavoring a party when creating district lines. Compactness refers to electoral districts remaining near each other. Contiguity refers to making sure districts are physically adjacent to each other.

## Problem Statement:

The main problem I am seeking to solve is limiting the deviation between districts to its absolute minimum to create the most balanced districts possible. I am only considering the raw population so demographics such as race, income, occupation will not factor into the plan.

## Model Description:

My model works mainly by minimizing the difference between the highest value allowed and the lowest value allowed and then constraining those to limits on the highs and lows. As well as limiting the model to only having one district per county/tract. This made sure no district had repeats from another district. I was able to get the best function minimizing from the graph with a larger number of tracts. This got me much closer to my ideal deviation.

# Model by Numbers:

My model works by using a simple binary type variable which factors in nodes through a json data file. This allows each county/tract to basically act as its own little "node" to grab data from such as name, geoID, population, or even demographic populations for each node. This gives a good amount of flexibility in the data. I had to run 2 models after I realized my initial model proposal wouldn't allow the data to split evenly into districts by the counties based on the way that the state is set up. My minimize function just consisted of u-l as largest-smallest. Each variable acted as the highest or lowest potential value that was allowed.

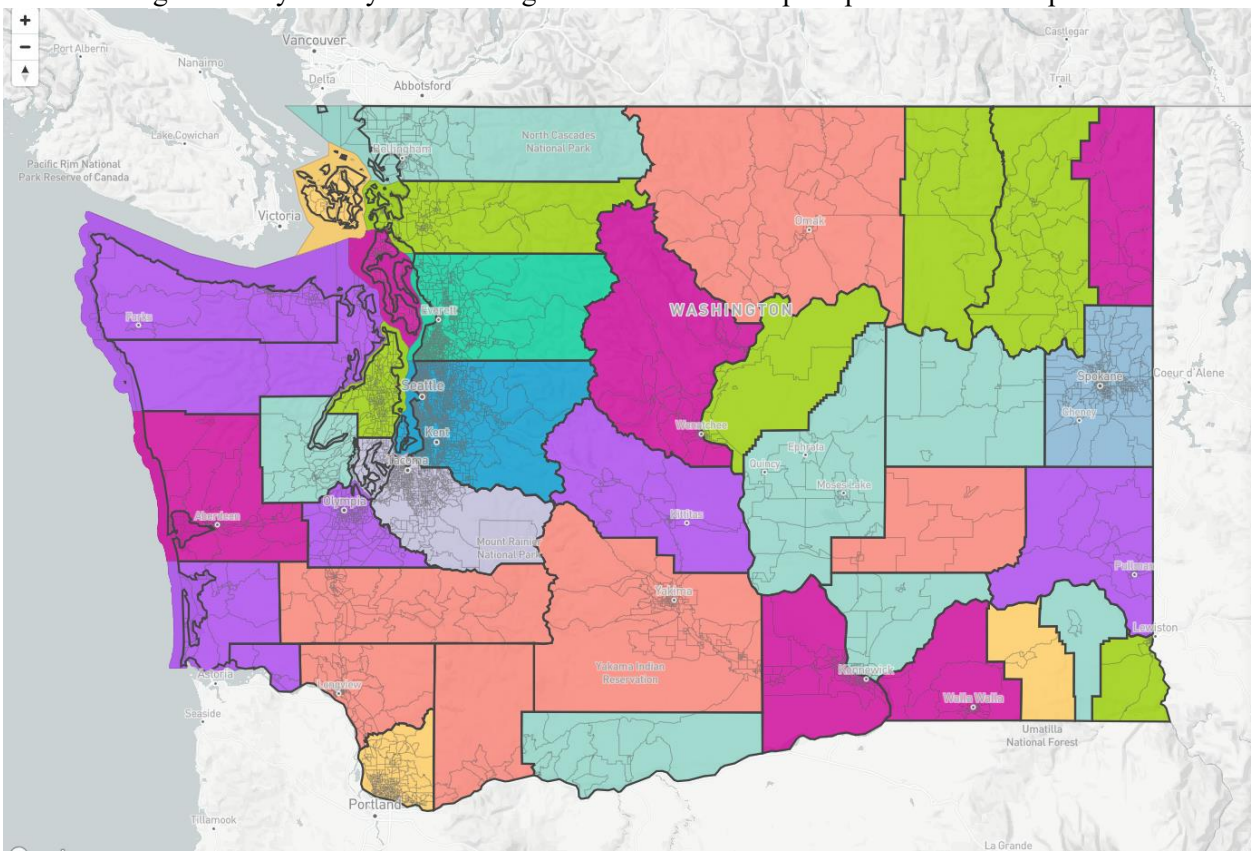
# Code:

My code was heavily based on another code from Buchanan (2020) but was altered to fit my state and my own objectives on the second model. Here is an example of the 1st model I built just to demonstrate how it was created.

```
m.setObjective(u-l, GRB.MINIMIZE) #minimizing the smallest difference between smallest and largest


#adding constraints

#adds constraint for 1 district per county

#each country should be kept whole for requirements

# add constraints saying that each county i is assigned to one district

m.addConstrs( gp.quicksum( x[i,j] for j in range(k) ) == 1 for i in G.nodes )


# add constraints saying that each district has population at least l

m.addConstrs( gp.quicksum( G.nodes[i]['TOTPOP'] * x[i,j] for i in G.nodes ) >= l for j in range(k) )


# add constraints saying that each district has population at most u

m.addConstrs( gp.quicksum( G.nodes[i]['TOTPOP'] * x[i,j] for i in G.nodes ) <= u for j in range(k) )


m.update()
```

# Plan/Maps:

I was only able to generate one map for my proposals as I was having an issue with data being difficult to transfer file types from .shp files to .json files. My first plan isn't very feasible as it was for my first model attempt where I used the counties as an attempt to separate the districts. Washington state has one county called King County which is significantly more populated than the rest of the counties in the state which makes values skewed. The second biggest issue with this plan is the fact that the districts do not follow legal scrutiny as they aren't contiguous. This was what prompted model attempt number 2.




# Evaluation of Plans:

Model 1 was very unfeasible as it lacked contiguity and lacked even a balanced population. This was mainly due to the constraints of the districts being locked in counties. For Model 2 it reaches a closer level of feasibility but lacked contiguity still. Model 2 used census tracts from the 2020 census instead of being limited to the counties in the state. This allowed the population to be spread much further out from each district. I was able to reach a deviation of 4441 people between the highest and lowest district. This was under the 1% recommendation from the NCLS. This did achieve my primary goal however this plan also still lacked contiguity so it is not a good plan for all the parameters that are required for a good redistricting plan. I believe my model could be further improved with more consistent data and a secondary set of constraints that help the model maintain contiguity. I also had the option of building a 3rd model using the census blocks instead of the census tracts which would further help the model achieve more population balance and allow further minimization of the objective

# Conclusions:

In conclusion, I developed two different model plans through a trial-and-error process for redistricting. My first model was ultimately a failure in setup as it prevented even distribution of population throughout the 10 congressional districts. My second model improved the population distribution while still lacking all the requirements for a good redistricting plan. I think a further model could improve the states districting even more.

# Citations:

Data: https://lykhovyd.com/files/public/districting/2020/WA/census_tracts/maps/

https://people.csail.mit.edu/ddeford/BLOCK/53.html\

https://fisherzachary.github.io/public/r-output.html#:~:text=Here%20we%20provide%20six%20of,)%20Length%2DWidth%20Ratio%20(C.C.

https://github.com/hamidrezavalidi/Political-Districting-to-Minimize-Cut-Edges

Map: https://districtr.org/plan

https://github.com/AustinLBuchanan/Districting-Examples-2020/tree/main

https://austinlbuchanan.github.io/OR-redistricting-resources/