

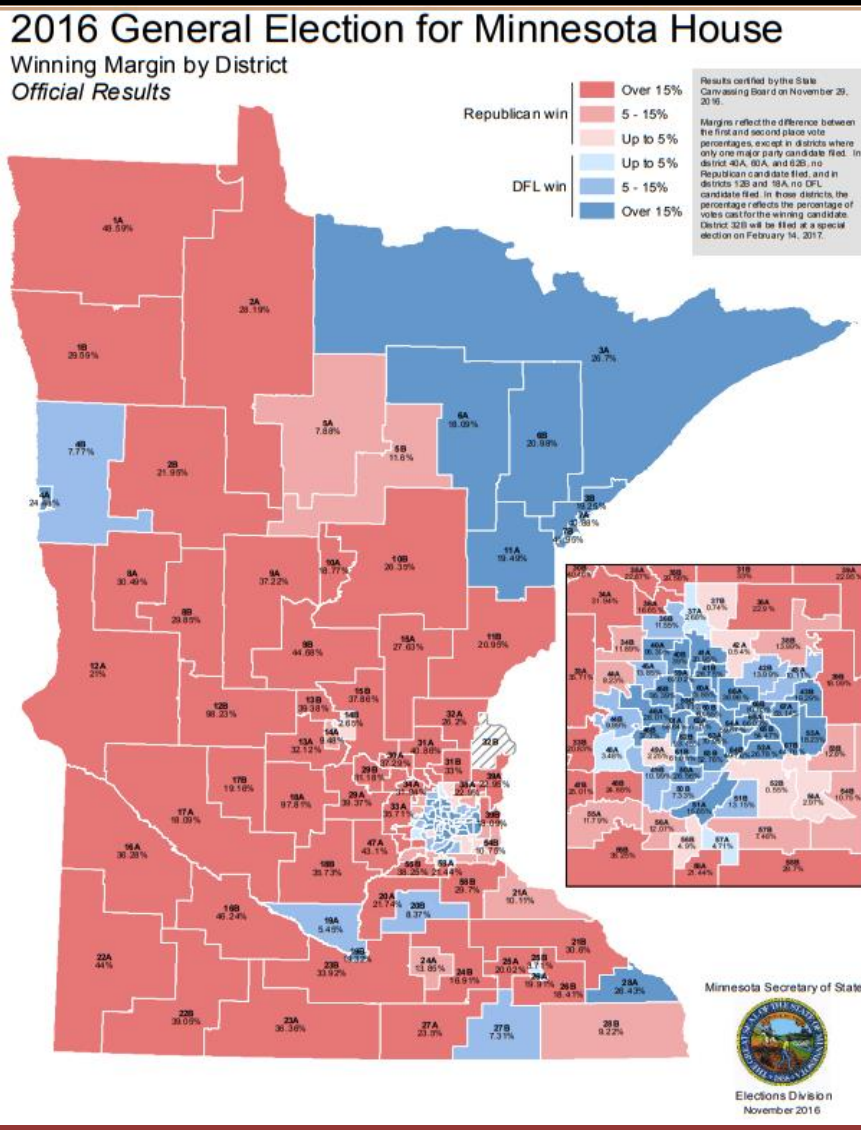
# Introduction

Gerrymandering is the drawing political boundaries to give party numeric advantage over an opposing party.

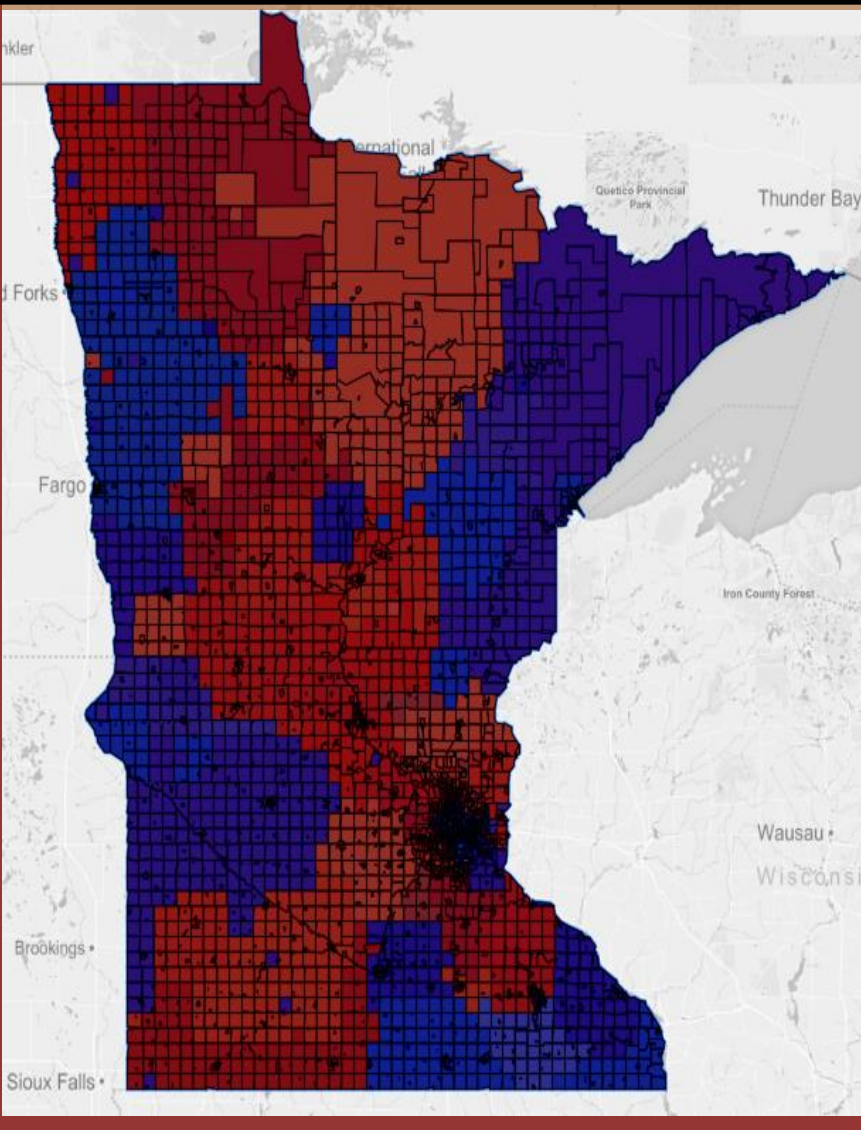
There are 87 counties and 8 districts in the state of Minnesota. This research poster is will be showing 4 elections result maps of U.S. Congress, Presidential, MN Senate, and MN House.

This project presents two maps, one with the current election district from 2016 and another with the redrawn district using algorithm to optimize compactness. The goal here is to compare the current and redrawn distircts and evaluate if there is any possibility of gerrymandering.

## MN House



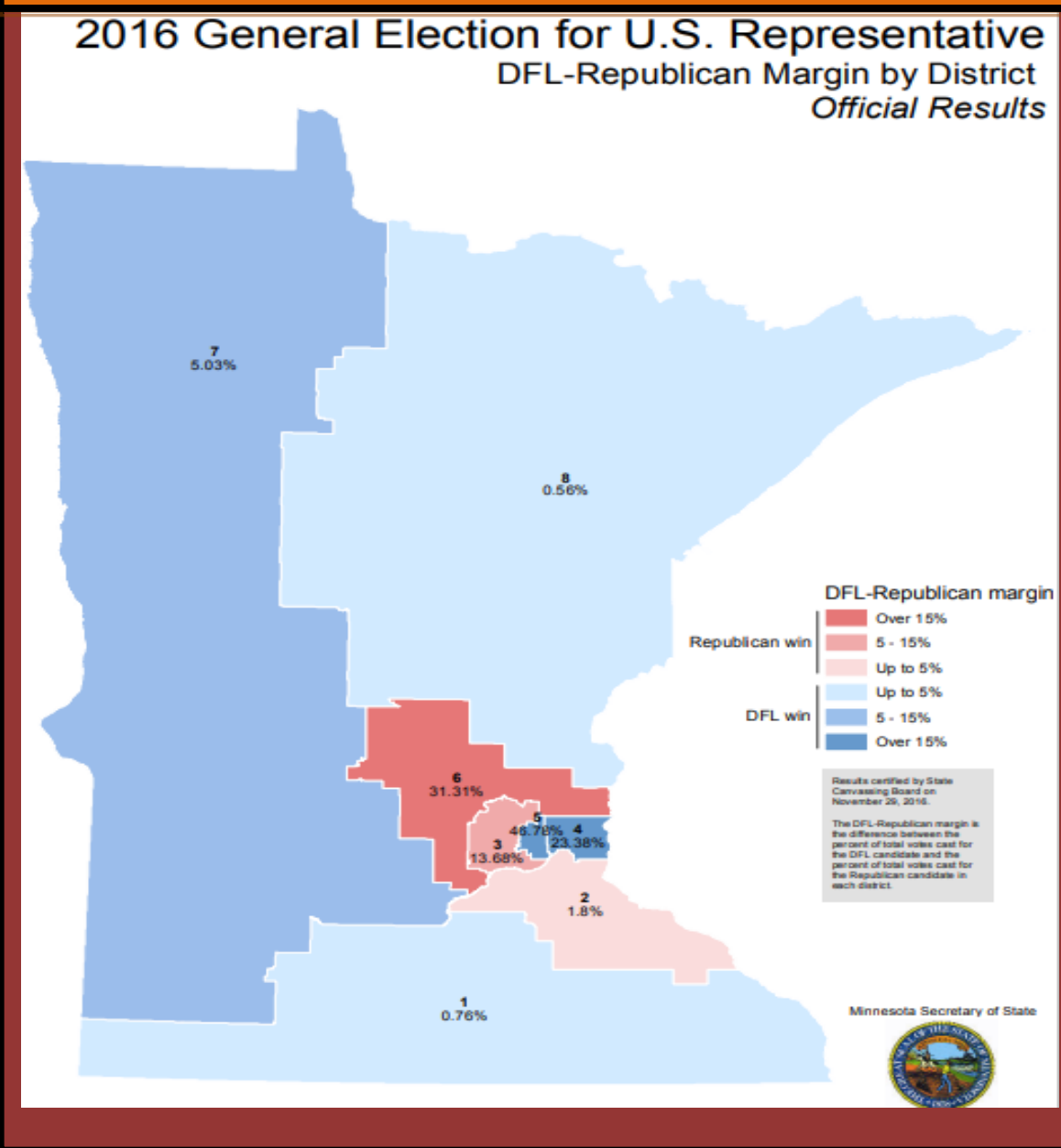
## MN House (Redrawn Map)



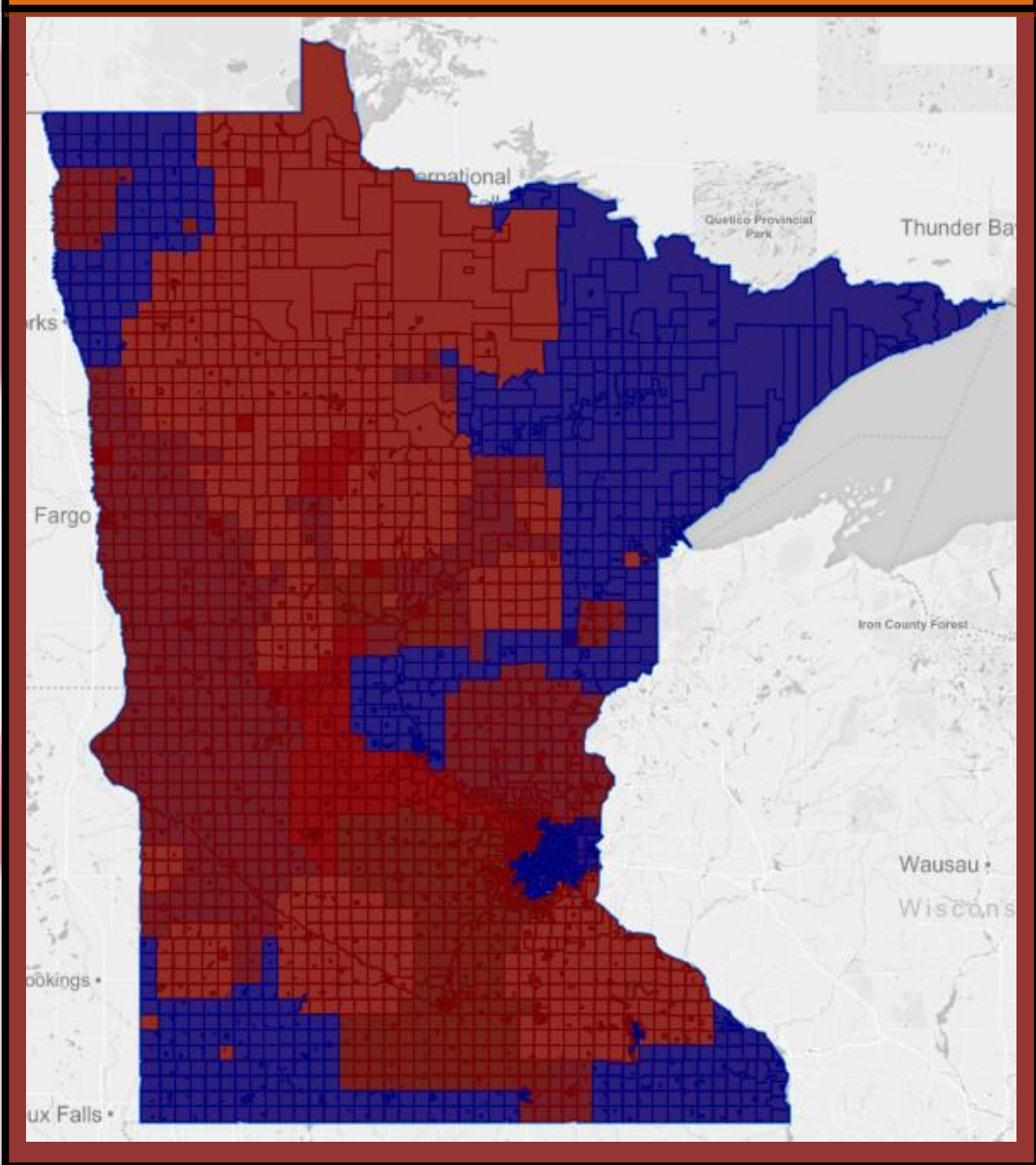
# GERRYMANDERING IN MINNESOTA

## 2016 Election Results

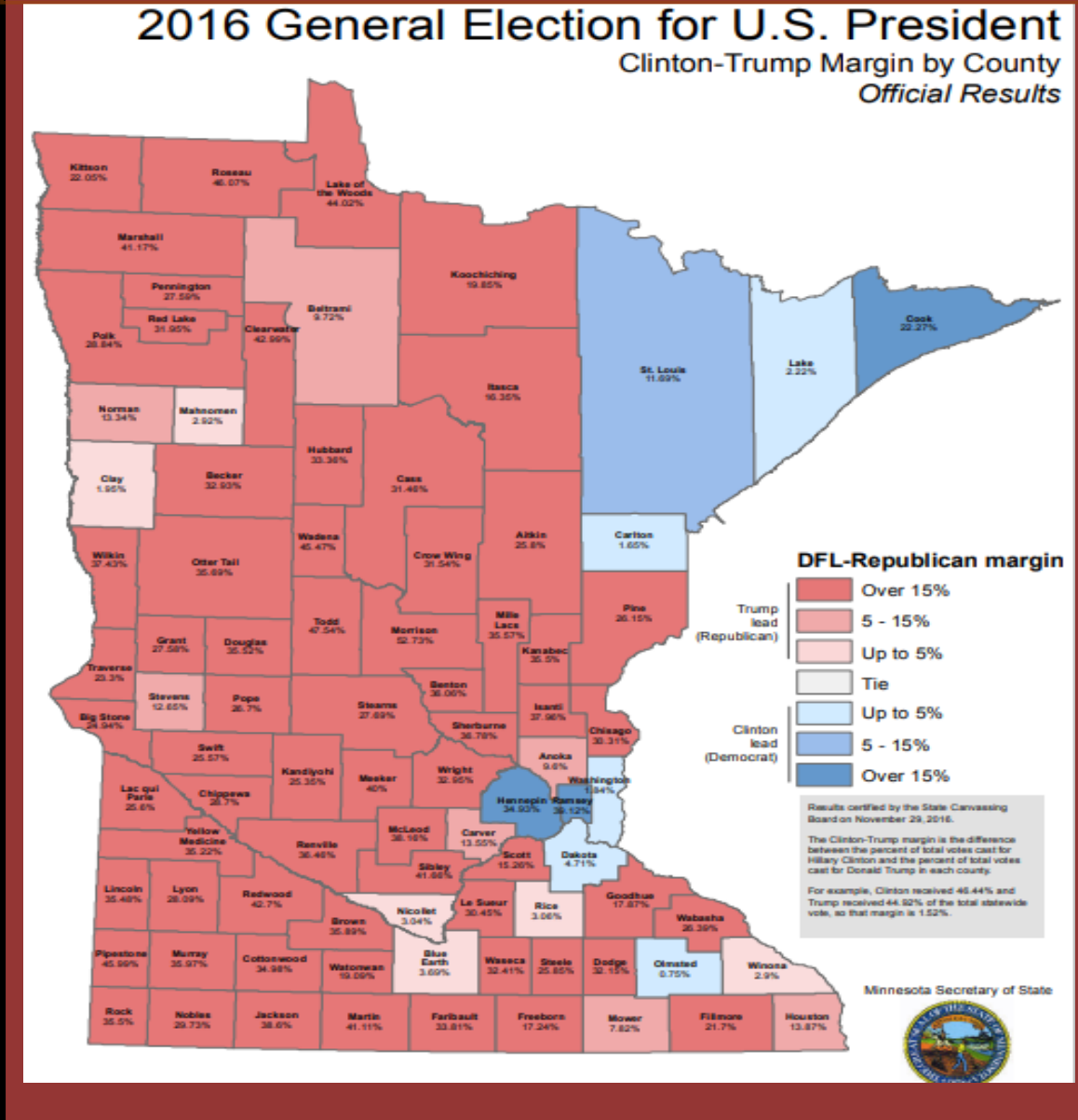
### U.S. Congress



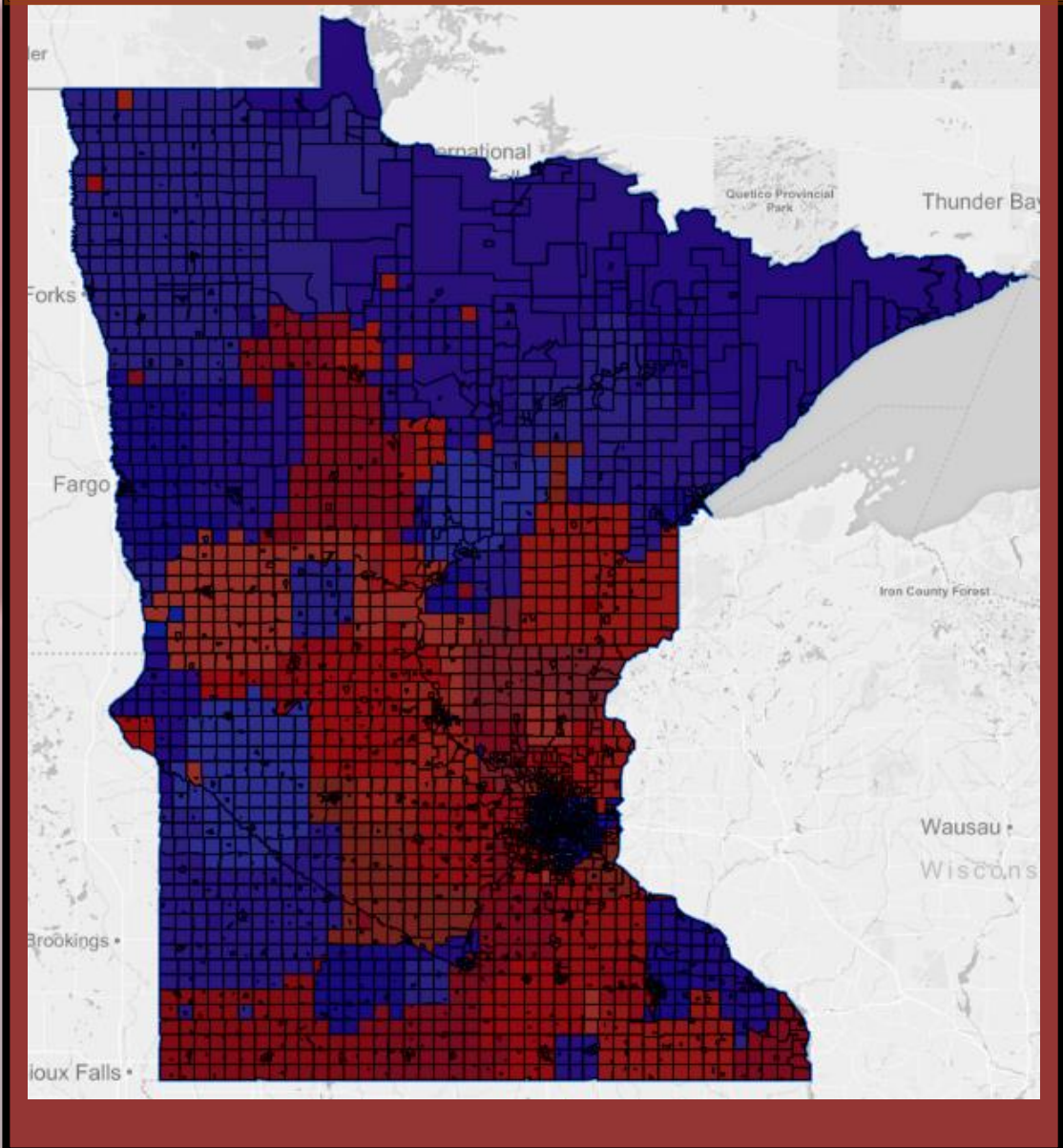
### U.S. Congress (Redrawn Map)



### Presidential



### Presidential (Redrawn Map)



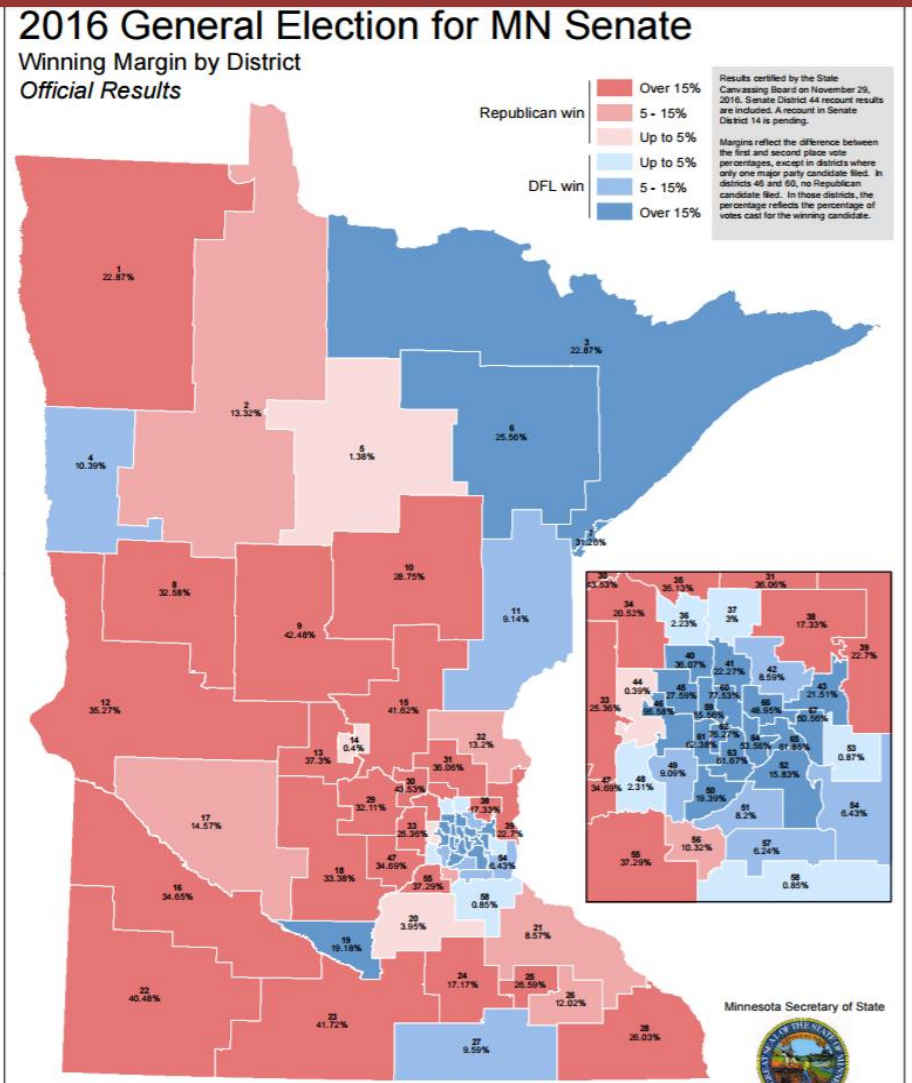
## Conclusion:

Gerrymandering project analyzes the precinct data from Minnesota State and compares the current election district and redrawn districts. This evaluates if there is any possibility of gerrymandering. The use of interactive map helps to visualize the data in a convenient way. There are two maps, current map and redrawn map for each district. Election results for each district can be viewed through the map. Current result show the election result from 2016 and redrawn district map optimizes compactness. This project shows how much influence can a party have over a region to win that area and the ways we can come up with using algorithm to maintain consistency and fairness in the election.

## Algorithms and Functionality

The redrawn map is created from a java project. The java Project has 3 main parts that alters the GeoJson file (This file list all the precincts and the latitude and longitude of all precincts and more). The first part looks at all the precincts in the GeoJson file and puts it into an Array List. It then looks at a precinct one by one and constantly searches the other precincts to find 2 of the same latitude and longitudes, this is how we get the adjacent precincts. The second part of this project was the toughest. First, we started off with a random precinct and added it into a group (can only hold totalPopulation/numberOfGroups). We then added all the adjacent precincts of the first one to the same group. From there we found a precinct from the adjacent precinct that had the least number of adjacent precincts. We then added all those adjacent precincts. If the group has not reached capacity and the precinct reaches the border of the map (an example of this is if we grouped the whole right side of the map but not yet grouped the left) we look for the precincts that have adjacent precincts that are not grouped, look for the precincts with the least amount of un grouped adjacent precincts and start from there. We do this until complete. Part 3 looks for precincts that were not grouped and add them to a group that surrounds them.

## MN Senate



## MN Senate (Redrawn Map)

