IE5311 Principles of Operation Research

New Mexico Districting Project

Team: NM Redistricting Team

Project members:

Bharath Billapati (R11861118)

Umamaheshwar rao Andiboina(R11847013)

Dinesh Nandan Koppaka(R11850676)

Executive Summary:

This project aims to create an optimal congressional district map for New Mexico using Python and optimization techniques. We developed two models following federal and state redistricting principles like equal population, contiguity, compactness, and minority representation.

The first model minimizes county boundary cuts, a common measure of compactness. The second minimizes district "spread" by treating counties as point masses and minimizing their inertia. Both models split New Mexico into three contiguous districts with under 1% deviation from perfect population equality.

We will present the project in three sections:

1) The mathematical concepts and logic behind the optimization approaches

2) Formal mathematical programs detailing the constraints that enforce redistricting laws and principles

3) Python code implementing the models

After obtaining solutions, we compare the models across criteria like population balance, compactness, and minority representation to select an optimal result.

The best model divides New Mexico into compact districts with small population deviations, respect for county boundaries, and sensitivity to minority groups. It has several advantages over current district maps that could promote fair political representation. This project illustrates how optimization techniques can create equitable legislative districts within legal constraints. Given gerrymandering concerns, tools like this could make the redistricting process more impartial. Our optimal 2020 map exemplifies how complex yet vital mapping rules seek to uphold political rights.

INTRODUCTION:

The constitutionally mandated ten-year nationwide census aims to count every single resident in the United States. This census, as outlined in Article I Section 2 of the Constitution, serves as the foundation for redrawing congressional district maps to reflect changes in each state's population over the previous decade. According to the United States Census Bureau, "redistricting is a critical process that aims to ensure everyone is equally represented" through this redistricting process based on the most recent demographic data.

Many experts, however, argue that politicians frequently use the post-census redistricting cycle to serve partisan interests rather than the needs of citizens. According to a Fair Vote report, modern redistricting often "encourages manipulation" as politicians gerrymander district boundaries to favor their allies, disfavor their opponents, and choose supportive voters rather than earning voters' support through their records. Gerrymandered maps distort the core democratic process by protecting incumbents and reducing electoral competition.

To fulfill the promise of fair representation rooted in the census count made by the Constitution, states must prioritize citizens over political parties when redrawing district maps. As New Mexico begins redistricting following the 2020 Census, it is critical that new district boundaries accurately reflect the state's changing population distribution, protect minority voting rights, and respect the distinct identities of New Mexican communities. Our project team has been tasked with creating an unbiased, optimization-based redistricting proposal for New Mexico's three congressional districts that meet both these representation goals and all federal and state legal requirements. We use the most recent census data, voting precinct geography, compactness metrics, and respect for county boundaries to design a plan that limits gerrymandering risks while also satisfying laws that occasionally conflict. This report details our transparent, mathematically based redistricting process that produced an equitable congressional map for the upcoming ten years of New Mexico democracy.

Current maps:

2010 Map: According to official government data, New Mexico was apportioned 3 congressional seats after the 2010 Census, which recorded a state population of 2,059,179. This resulted in an ideal district size of 686,393 residents based on equal apportionment. The enacted 2011 redistricting plan achieved perfect population equality across the three districts with a 0.0% deviation from the ideal. However, the map has been criticized for failing to sufficiently represent New Mexico's distinctive demographic and geographic diversity. As seen in the 2011 congressional map below, the district boundaries divide the northern and southern regions rather than reflecting distinct urban and rural

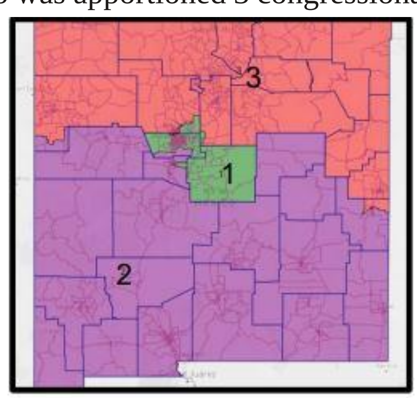


Figure 1.1: 2010 New Mexico Congressional Map

communities of interest. The serpentine, interconnected shapes also undermine compactness. As New Mexico embarks on another round of redistricting in 2023, there is considerable room for improvement over the prior map based on public input solicited by the Citizen Redistricting Committee. While equal population will remain a priority, more competitive districts that elevate minority voices and respect county boundaries would better serve popular representation.

2020 Map: According to the 2020 Census, New Mexico's total population rose to 2,117,522 residents, while still retaining its 3 seats in the U.S. House of Representatives. This yielded an ideal district size of 705,841 based on equal apportionment. On December 16, 2022, Governor Michelle Lujan Grisham approved a new congressional map with strong adherence to the principle of population equality. As depicted in the current 2022 map below, the total deviation in district population is just 14 persons above or below the ideal, amounting to only a 0.00066% overall deviation.

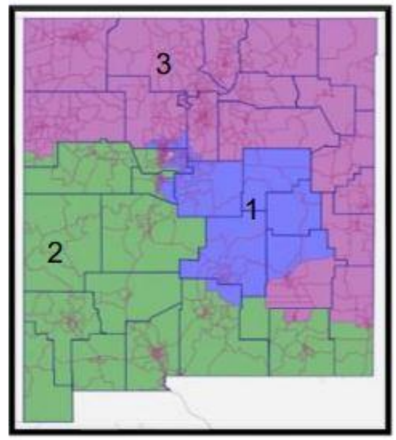


Figure 1.2: 2021 New Mexico Congressional Map

In a statement upon signing, Governor Grisham contended the new boundaries establish a reasonable baseline for electoral competitiveness, preventing any unfair partisan slant. However, some analysts note the map amalgamates urban and rural areas rather than preserving communities of interest. Others critique the irregular, sprawling shapes of the districts on compactness grounds. On the other hand, the 2nd District incorporates a major share of the state's Hispanic community, enhancing minority representation. Compared to past New Mexico redistricting cycles, the approved 2022 map largely fulfills the key criteria of equal population and minority voting rights, though it leaves room for improvement on respecting geographic compactness and communities of interest.

Criteria:

A congressional districting map must meet several national and state requirements to be accepted. The National Conference of State Legislatures (NCSL) website has these requirements. The sections that follow provide an outline of the criteria.

Federal Redistricting Requirements

The U.S. Constitution and subsequent federal legislation establishes baseline standards for drawing congressional districts during each decennial redistricting cycle. As interpreted over time by federal courts and the Justice Department:

Equal Population: Article I, Section 2 requires states to apportion House districts based on population, with the districts as equal in population "as practicable." While perfect mathematical equality is not expected, deviations generally cannot exceed 1-2% without justification tied to other legitimate policy objectives. The 2022 New Mexico map exemplifies this standard with a minuscule 0.00066% deviation.

Minority Voting Rights: The 14th Amendment's equal protection clause together with Section 2 of the Voting Rights Act of 1965 prohibits racial gerrymandering and vote dilution of historically underrepresented groups. States must ensure people of color have equal opportunity to elect preferred candidates in a number of "opportunity" districts reflective of their share of the population. New Mexico's 2nd District in the latest plan achieves this aim.

Other Principles: Although factors like compactness, contiguity, and preservation of political subdivisions are not federally mandated, over half of states incorporate these standards in their redistricting process. Commonly adopted criteria also include keeping communities of interest intact where possible and avoiding unnecessary contests between incumbents. However, federal law supersedes any conflicting state provisions.

In sum, adherence to equal population requirements and protection of minority voting power form the basis of a legally valid congressional map. While competitiveness and state-specific districting principles may supplement federal guidelines, they cannot undermine guarantees of representation enshrined in the Constitution and Voting Rights Act. New Mexico's 2022 congressional plan largely aligns with key federal priorities around population equality and minority voting rights.

New Mexico Redistricting Requirements

In addition to federal standards, New Mexico statutes and legislative guidelines impose state-level prerequisites for new congressional maps centered around four main principles:

Compactness & Contiguity

Districts must comprise geographically compact areas made up of contiguous precincts to the extent possible. The 2022 New Mexico map's irregular, sprawling district shapes fall short on compactness measures.

Communities of Interest

Boundaries should preserve shared communities of interest such as racial/ethnic groups, cultural regions, metropolitan areas, and economic nodes united by common interests and issues. The 2022 plan draws some criticism for dividing such communities.

Political Subdivisions

Counties, municipalities, and other existing political subdivisions and geographic areas should not be split where avoidable. The latest New Mexico map crosses several county lines unnecessarily according to opponents.

Neutral Districting

Plans should uphold nonpartisan traditional redistricting principles without unduly considering incumbents' residences or partisan political data. However, the 2022 map faces accusations of Democratic gerrymandering by amalgamating urban and rural regions.

While New Mexico officially adheres to these state standards, the new 2022 congressional plan has sparked debate over whether they were adequately fulfilled, especially regarding compactness, communities of interest, county integrity, and partisan neutrality. Potential areas for improvement remain. Nonetheless, the map aligns with the equally binding state requirements around equal population and minority representation through its precisely balanced districts and majority-Hispanic 2nd District.

Problem Statement:

The core goal guiding our redistricting approach is to produce a statistically optimized three-district congressional map for New Mexico that maximizes compactness as measured by analytical district shape metrics. We will leverage computational optimization tools to impartially engineer district boundaries without any explicit partisan or incumbent-related considerations, avoiding issues plaguing prior politicized maps.

Subject to this overarching focus on maximal compactness, our algorithmically generated redistricting plans will incorporate constraints to fulfill all binding federal and state legal criteria. This includes strict adherence to equal population requirements as well as protection of minority voting rights through at least one Hispanic-majority opportunity district. We will also respect New Mexico's standards around preserving whole counties given their importance as administrative units containing distinct communities of interest.

Additionally, we plan to thoroughly vet our optimized computer-drawn maps to ensure they contain: 1) Three reasonably compact, contiguous districts comprised of whole precincts, 2) Districts that uphold county integrity except when unavoidable, and 3) Equitable minority representation where all communities have a voice. By leveraging the flexibility of computing power subject to fairness-focused constraints, we aim to provide an impartial New Mexico redistricting proposal for consideration to replace the controversial 2022 gerrymandered map.

OR Model (in words)

To tackle the intricacies of redistricting, our team has formulated two optimization models, each with a unique focus. Model 1 centers around the minimization of cut edges, strategically working to reduce boundaries between counties and enhance the compactness of districts. On the other hand, Model 2 is dedicated to minimizing the moment of inertia, treating each district as a rigid body with properties influenced by population and spatial distribution. Both models are meticulously crafted to adhere to essential constraints, ensuring population balance and geographic contiguity. This dual-model approach allows us to explore distinct strategies for creating fair, representative, and well-connected congressional districts, presenting a comprehensive solution to the multifaceted challenges of redistricting.

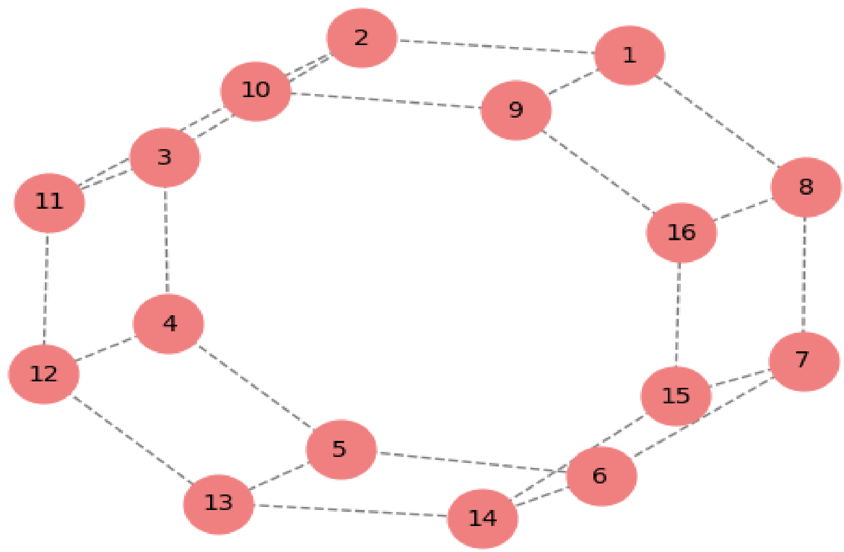
Minimize Cut Edges (Model 1):

In the pursuit of constructing an optimal congressional districting model, Model 1 emerges as a powerful tool with the overarching objective of minimizing cut edges. These "cut edges" delineate the boundaries between two counties and hold significant weight in the context of congressional districting, representing the lines that separate counties into distinct political districts. The crux of Model 1 lies in the strategic reduction of these cut edges, a task that goes beyond mere cartography—it involves shaping political landscapes.

To ensure the model's validity, a set of rigorous constraints are meticulously integrated. Each county is designated to a singular district, fostering clarity and preventing ambiguities in representation. Population distribution across districts becomes a critical consideration, with the ideal scenario characterized by equal populations in each district and a minimal acceptable deviation, typically within ± 0.5% of the target value. This pursuit of demographic equilibrium is fundamental to the principles of fair representation.

Contiguity, the uninterrupted connection of neighboring areas, is a linchpin in the efficacy of congressional districts. The model enforces constraints that mandate every district to form a cohesive, continuous unit. Within this construct, districts are conceptualized with a "root" or center, and contiguity is realized by facilitating the flow across edges that are not marked as cut edges. Through these measures, the model seeks to guarantee that every county within a congressional district is an integral part of a unified flow from the root—an essential element in fostering a coherent and interconnected representation of diverse communities.

In essence, Model 1 transcends the conventional understanding of districting by integrating mathematical precision, geographic insight, and demographic equity. It stands as a testament to the nuanced interplay between political representation and spatial configuration, offering a methodology that strives for the delicate balance between minimizing cut edges and upholding the principles of fair and representative democracy.



The figure represents a conceptual map used in the context of congressional districting, specifically in the application of Model 1, where the primary objective is to minimize cut edges. Here's a breakdown of the key components:

Nodes: The circles in the diagram represent nodes, each of which signifies a county or a geographic entity. In this scenario, we have 16 nodes labeled from 1 to 16.

Edges: The gray dashed lines between nodes represent edges, which are boundaries between counties. These edges play a crucial role in defining the districts.

Cut Edges: The gray dashed lines specifically highlight cut edges, indicating boundaries across which two nodes (counties) are separated into different regions or districts. The minimization of these cut edges is the focus of Model 1 in the congressional districting process.

District Division Strategies: The diagram illustrates two different strategies for dividing the 16 nodes (counties) into districts. On the left, the nodes are divided into four columns, resulting in 12 cut edges. On the right, the nodes are divided into four squares, leading to only eight cut edges. The goal is to strategically organize the nodes to minimize cut edges and, consequently, create more compact regions.

Compact Regions: Generally, when a set of nodes has fewer cut edges, it produces more compact regions. Compactness is a key factor in creating effective and fair congressional districts, and it is achieved by minimizing the fragmentation caused by cut edges.

In the broader context of congressional districting, this diagram visually represents the challenge of dividing geographic entities into districts while minimizing the disruption caused by cut edges. The goal is to design districts that are both compact and contiguous, ensuring a fair and representative distribution of population. The strategies employed in this diagram lay the groundwork for the optimization models used in the redistricting process.

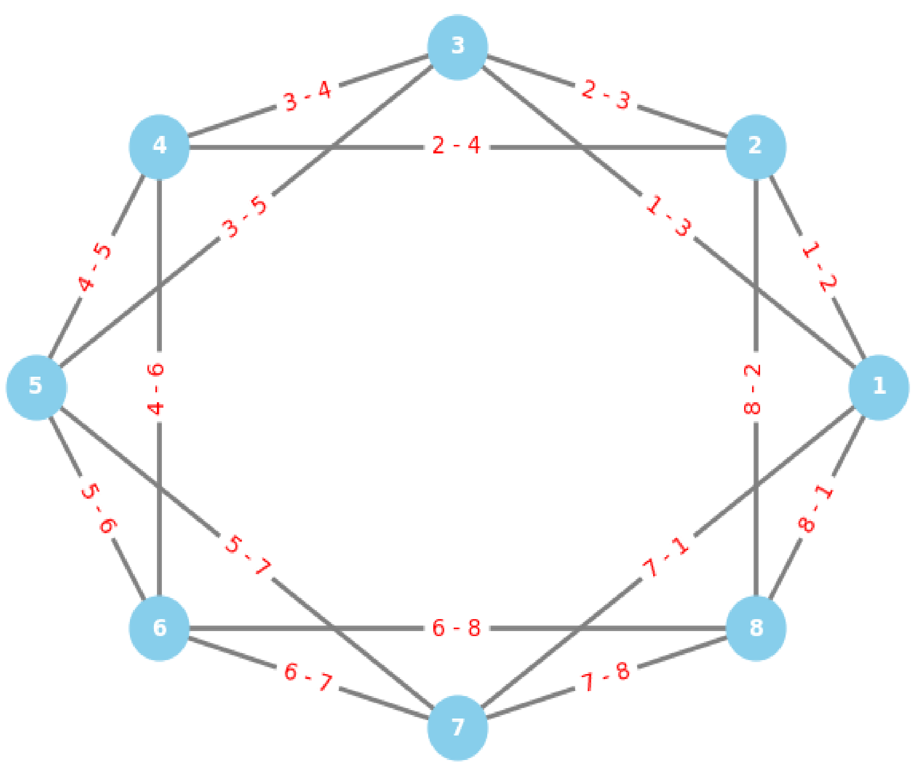
Minimize Moment of Inertia (Model 2):

In the Minimize Moment of Inertia model (Model 2), the focus shifts towards optimizing the redistricting process by leveraging the physical concept of rotational inertia. This model treats each congressional district as a rigid body, where the 'mass' is indicative of the district's population, and the spatial arrangement of counties influences the moment of inertia. Unlike the Cut Edges approach, this model introduces a unique perspective, drawing parallels between rotational motion principles and redistricting efficiency.

The moment of inertia is computed by summing the product of each county's population and the squared distance between counties. By minimizing this moment of inertia, the model seeks to achieve a redistricting plan that not only adheres to essential constraints—such as each county belonging to a single district, maintaining population equality, and ensuring district continuity—but also enhances the overall spatial coherence of the districts.

This approach combines mathematical precision with a physical analogy to address the intricate challenges of redistricting, balancing both population distribution and geographical compactness. The Minimize Moment of Inertia model, alongside the Cut Edges model, represents a comprehensive strategy to develop optimal and fair congressional districting solutions, with their respective merits and drawbacks to be thoroughly explored in subsequent analyses.

Minimize Moment of Inertia (Model 2) Diagram



The Minimize Moment of Inertia (Model 2) diagram represents a conceptual network used in the context of congressional districting. In this model, the objective is to minimize the moment of inertia, a measure of rotational inertia analogous to mass in translational motion. Each node in the graph symbolizes a county, and the edges between nodes signify the connections or boundaries between these counties.

The moment of inertia is calculated based on the population of each county and the distance between them. The larger the population (mass), the higher the moment of inertia, representing increased resistance to change. The distance between counties also influences the moment of inertia, establishing a linear relationship between inertia and the spatial arrangement of counties.

The graph is structured to resemble a circular layout for clarity. Each node is labeled with its corresponding county, and the edges are labeled to indicate the connections between counties. The color-coded nodes and edges contribute to the visual representation, aiding in the understanding of the network's structure.

The Minimize Moment of Inertia model aims to create congressional districts with optimized compactness and spatial coherence. By strategically arranging counties to minimize the moment of inertia, the model seeks to enhance the overall efficiency and integrity of the redistricting plan. This approach combines mathematical precision with geographical considerations to address the complex task of balancing population distribution and geographic compactness in congressional districting.

OR Model (In math):

- **Minimize Cut Edges (Model 1):**

The model focuses on minimizing cut edges, ensuring county assignment to districts, maintaining population balance, enforcing contiguity, and considering district compactness.

Parameters:

Xij = Binary variable indicating whether county *'i'* is in district *'j'*.

U, L: Upper and lower bounds for district populations.

Decision Variables:

$\boldsymbol{Xij}$: Binary variable indicating whether county *'i'* is in district *'j'*.

$\boldsymbol{Y_e}$: Binary variable indicating whether edge *e* is a cut edge.

Objective Function:

$$\text{Minimize} \quad \sum_{e \in E} y_e$$

Constraints:

1. **County Assignment:**
   $\sum_j x_{ij} = 1, \quad \forall i \in V$
2. **Population Bounds:**
   $L \leq \sum_i P_i \cdot x_{ij} \leq U, \quad \forall j$
3. **Contiguity and Flow:**
   $x_{ij} - x_{ik} \leq M \cdot y_{ek}, \quad \forall (i,k) \in E, \forall j$
   $\sum_k x_{ik} = 1, \quad \forall i \in V$
   $\sum_k x_{kj} - \sum_k x_{ik} = 0, \quad \forall i, j$
4. **Compactness:**
   $y_e = \frac{1}{2}(x_{ik} - x_{jk} + x_{jk} - x_{ik}), \quad \forall e = (i,j) \in E, \forall k \neq j$

Binary Constraints:

$x_{ij}, y_e \in \{0, 1\}$

This model aims to minimize the number of cut edges while satisfying constraints related to county assignment, population bounds, contiguity, and compactness. It provides a formal representation of the described optimization problem in congressional districting.

- **Minimize Moment of Inertia (Model 2):**

The model focuses on minimizing the moment of inertia while adhering to constraints related to county assignment, population balance, district continuity, and spatial coherence

Parameters:

Xij = Binary variable indicating whether county *'i'* is in district '*j'*.

Objective Function:

$$\text{Minimize} \quad \sum_{i,k \in V} P_i \cdot P_k \cdot (d_{ik})^2 \cdot x_{ij} \cdot x_{kj}$$

Constraints:

1. **County Assignment:**
$\sum_j x_{ij} = 1, \quad \forall i \in V$
2. **Population Bounds:**
$L \leq \sum_i P_i \cdot x_{ij} \leq U, \quad \forall j$
3. **District Continuity:**
$\sum_k x_{ik} - \sum_k x_{jk} = 0, \quad \forall i, j$
4. **Moment of Inertia:**
$\sum_{i,k \in V} P_i \cdot P_k \cdot (d_{ik})^2 \cdot x_{ij} \cdot x_{kj} \leq M$

Binary Constraints:
$x_{ij} \in \{0, 1\}$

This model aims to minimize the moment of inertia while satisfying constraints related to county assignment, population bounds, district continuity, and the overall spatial arrangement of counties. It provides a formal representation of the described optimization problem in congressional districting using the Minimize Moment of Inertia approach.

PYTHON/GUROBI CODE:

Minimum Cut Edges:

- Read Graph:

A New Mexico county graph is read from a JSON file using the 'gerrychain' library.

```
In [2]: from gerrychain import Graph
```

```
In [3]: # Read New Mexico county graph from the json file "NM_county.json"
filename = 'NM_county.json'

# GerryChain has a build-in function for reading graphs of this type:
G = Graph.from_json(filename)

# Print the nodes
print("The New Mexico county graph has nodes =", G.nodes)

The New Mexico county graph has nodes = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32]
```

```
In [4]: # Let's impose a 1% population deviation (+/- 0.5%)
deviation = 0.01

import math
k = 3         # number of districts


# For each node, print the node #, county name, and its population
for node in G.nodes:
    county_name = G.nodes[node]['NAME20']
    county_population = G.nodes[node]['P0010001']
    G.nodes[node]['TOTPOP'] = county_population
    print("Node", node, "represents", county_name, "County which had a population of", county_population,"in the 2020 cencus.")

total_population = sum(G.nodes[node]['TOTPOP'] for node in G.nodes) # Population of New Mexico

print("\nThe total population of New Mexico is", total_population)

L = math.ceil((1-deviation/2)*total_population/k)  # Lower Bound of District Population
U = math.floor((1+deviation/2)*total_population/k) # Upper Bound of District Population

print("\nUsing L =",L,"and U =",U,"and k =",k)
```

- Population Data:

Population data for each county in the graph is retrieved and printed.

Total population of New Mexico is calculated.

```
In [5]: import gurobipy as gp
from gurobipy import GRB

# create model
m = gp.Model()

# create variables
x = m.addVars(G.nodes, k, vtype=GRB.BINARY) # x[i,j] equals one when county i is assigned to district j
y = m.addVars(G.edges, vtype=GRB.BINARY)   # y[u,v] equals one when edge {u,v} is cut
```

```
In [6]: import gurobipy as gp
from gurobipy import GRB

# create model
m = gp.Model()

# create variables
x = m.addVars(G.nodes, k, vtype=GRB.BINARY) # x[i,j] equals one when county i is assigned to district j
y = m.addVars(G.edges, vtype=GRB.BINARY)   # y[u,v] equals one when edge {u,v} is cut
```

```
In [5]: # objective is to minimize cut edges
m.setObjective( gp.quicksum( y[u,v] for u,v in G.edges ), GRB.MINIMIZE )
```

```
In [6]: # add constraints saying that each county i is assigned to one and only 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 must have a population between L and U
m.addConstrs( gp.quicksum( G.nodes[i]['TOTPOP'] * x[i,j] for i in G.nodes) >= L for j in range(k) )
m.addConstrs( gp.quicksum( G.nodes[i]['TOTPOP'] * x[i,j] for i in G.nodes) <= U for j in range(k) )

# add constraints saying that edge {i,j} is cut if i is assigned to district v but j is not.
m.addConstrs( x[i,v] - x[j,v] <= y[i,j] for i,j in G.edges for v in range(k))

m.update()
```

- Define Problem Parameters:

Deviation, the allowed percentage population deviation, is set.

The number of districts (k) is set.

Lower and upper bounds for district populations are calculated.

```
In [7]: # CONTIGUITY CONSTRAINTS!!!
        # We will use the contiguity constraints of Hojny et al. (MPC, 2021)
        #   https://link.springer.com/article/10.1007/s12532-020-00186-3

        # Add root variables: r[i,j] equals 1 if node i is the "root" of district j
        r = m.addVars(G.nodes, k, vtype=GRB.BINARY)

        # Add flow variables: f[u,v] = amount of flow sent across arc uv
        #  Flows are sent across arcs of the directed version of G which we call DG
        import networkx as nx
        DG = nx.DiGraph(G) # directed version of G
        f = m.addVars(DG.edges, vtype=GRB.CONTINUOUS)
```

```
In [8]: # The big-M proposed by Hojny et al.
        M = G.number_of_nodes() - k + 1

        # Each district j should have one root
        m.addConstrs( gp.quicksum( r[i,j] for i in DG.nodes) == 1 for j in range(k) )

        # If node i is not assigned to district j, then it cannot be its root
        m.addConstrs( r[i,j] <= x[i,j] for i in DG.nodes for j in range(k) )

        # if not a root, consume some flow.
        # if a root, only send out (so much) flow.
        m.addConstrs( gp.quicksum( f[u,v] - f[v,u] for u in DG.neighbors(v) ) >= 1 - M * gp.quicksum( r[v,j] for j in range(k)) for v in G.nodes)

        # do not send flow across cut edges
        m.addConstrs( f[i,j] + f[j,i] <= M * (1 - y[i,j]) for (i,j) in G.edges )

        m.update()
```

- Gurobi Model Setup:

Gurobi optimization model (m) is created.

Binary decision variables (x and y) are added to the model.

Objective function is set to minimize the number of cut edges.

```
Cutting planes:
  Gomory: 4
  Cover: 1
  Implied bound: 10
  MIR: 23
  StrongCG: 1
  Flow cover: 1
  Zero half: 2
  RLT: 25

Explored 660 nodes (17196 simplex iterations) in 0.69 seconds (0.39 work units)
Thread count was 8 (of 8 available processors)

Solution count 2: 19 21

Optimal solution found (tolerance 1.00e-04)
Best objective 1.900000000000e+01, best bound 1.900000000000e+01, gap 0.0000%
```

- Add Constraints:

Constraints are added to ensure that each county is assigned to exactly one district.

Constraints are added to ensure that each district's population falls within the specified bounds.

Constraints are added to identify cut edges.

- Contiguity Constraints:

Root variables (r) and flow variables (f) are added.

Hojny et al.'s contiguity constraints are implemented.

- Solve the Model:

The Gurobi optimization model is solved.

The number of cut edges in the optimal solution is printed.

```
In [10]: print("The number of cut edges is",m.objval)

# retrieve the districts and their populations
districts = [ [i for i in G.nodes if x[i,j].x > 0.5] for j in range(k)]
district_counties = [ [ G.nodes[i]["NAME20"] for i in districts[j] ] for j in range(k)]
district_populations = [ sum(G.nodes[i]["TOTPOP"] for i in districts[j]) for j in range(k) ]

# print district info
for j in range(k):
    print("District",j,"has population",district_populations[j],"and contains counties",district_counties[j])

# What is the population deviation for this model?
population_deviation = max(district_populations) - min(district_populations)
print("\nThe population deviation for this model is", population_deviation, "people.")
```

```
The number of cut edges is 19.0
District 0 has population 702632 and contains counties ['Curry', 'San Juan', 'Guadalupe', 'McKinley', 'Rio Arriba', 'Santa Fe', 'Quay', 'Union', 'San Miguel', 'Taos', 'Colfax', 'Sandoval', 'Harding', 'Los Alamos', 'Mor
a']
District 1 has population 707195 and contains counties ['Bernalillo', 'Catron', 'Cibola']
District 2 has population 707695 and contains counties ['Roosevelt', 'Lincoln', 'Torrance', 'Luna', 'Lea', 'Chaves', 'Sierra', 'Valencia', 'De Baca', 'Otero', 'Grant', 'Doña Ana', 'Hidalgo', 'Socorro', 'Eddy']

The population deviation for this model is 5063 people.
```

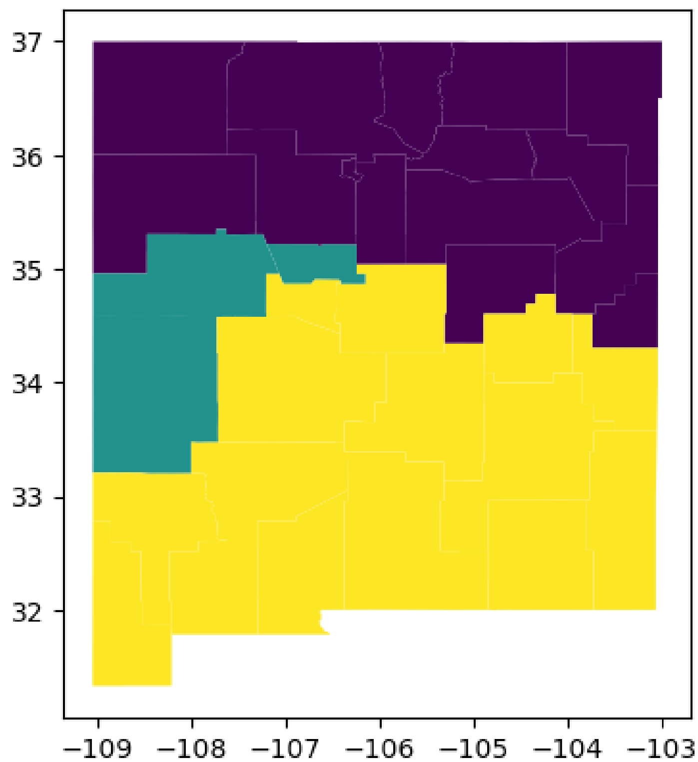
- Retrieve District Information:

Districts, their counties, and populations are extracted from the model.

- Population Deviation:

Population deviation among districts is calculated.

- Map Results:



Geopandas is used to read a shapefile of New Mexico counties.

Counties are assigned to districts based on the optimization results, and a map is generated.

- Results Summary:

The number of cut edges, population distribution among districts, and a map illustrating district assignments are printed.

In summary, the code demonstrates the use of Gurobi for solving a districting problem, considering population constraints and contiguity, and visualizing the results on a map.

Minimize Moment of Inertia (Model 2):

In [1]:
```
from gerrychain import Graph
```

In [2]:
```
# Read New Mexico county graph from the json file "NM_county.json"
filename = 'NM_county.json'

# GerryChain has a build-in function for reading graphs of this type:
G = Graph.from_json(filename )
```

In [3]:
```
# Find the name, population, longitude, and latitude of each county

for node in G.nodes:
    name = G.nodes[node]['NAME20']                          # Name of county
    county_population = G.nodes[node]['P0010001']          # Populaton of county
    G.nodes[node]['TOTPOP'] = county_population
    G.nodes[node]['C_X'] = G.nodes[node]['INTPTLON20'] # LONGITUDE of county center
    G.nodes[node]['C_Y'] = G.nodes[node]['INTPTLAT20'] # LATITUDE of county center
    x_coordinate = G.nodes[node]['C_X']
    y_coordinate = G.nodes[node]['C_Y']

    # Output the information
    print("Node", node, "is", name, "County, which has population", county_population, "and is centered at (",x_coordinate,",",y_coordinate,")")
```

```
Node 0 is Curry County, which has population 48430 and is centered at ( -103.3460546 , +34.5729841 )
Node 1 is Roosevelt County, which has population 19191 and is centered at ( -103.4830039 , +34.0212068 )
Node 2 is San Juan County, which has population 121661 and is centered at ( -108.3245778 , +36.5116245 )
Node 3 is Lincoln County, which has population 20269 and is centered at ( -105.4498055 , +33.7408411 )
Node 4 is Guadalupe County, which has population 4452 and is centered at ( -104.7849677 , +34.8697822 )
Node 5 is Torrance County, which has population 15045 and is centered at ( -105.8905574 , +34.5549784 )
Node 6 is Luna County, which has population 25427 and is centered at ( -107.7471911 , +32.1845231 )
Node 7 is McKinley County, which has population 72902 and is centered at ( -108.2532938 , +35.5840616 )
Node 8 is Rio Arriba County, which has population 40363 and is centered at ( -106.6939829 , +36.5096687 )
Node 9 is Lea County, which has population 74455 and is centered at ( -103.4132707 , +32.7956865 )
Node 10 is Chaves County, which has population 65157 and is centered at ( -104.4698374 , +33.3616045 )
Node 11 is Santa Fe County, which has population 154823 and is centered at ( -105.9639718 , +35.5145309 )
Node 12 is Quay County, which has population 8746 and is centered at ( -103.5480713 , +35.1070184 )
Node 13 is Bernalillo County, which has population 676444 and is centered at ( -106.6690805 , +35.0536280 )
Node 14 is Sierra County, which has population 11576 and is centered at ( -107.1881612 , +33.1194684 )
Node 15 is Valencia County, which has population 76205 and is centered at ( -106.8065821 , +34.7168404 )
Node 16 is De Baca County, which has population 1698 and is centered at ( -104.3686961 , +34.3592729 )
Node 17 is Otero County, which has population 67839 and is centered at ( -105.7513079 , +32.6155988 )
Node 18 is Catron County, which has population 3579 and is centered at ( -108.3919284 , +33.9016208 )
Node 19 is Union County, which has population 4079 and is centered at ( -103.4757229 , +36.4880853 )
Node 20 is San Miguel County, which has population 27201 and is centered at ( -104.8035189 , +35.4768585 )
Node 21 is Taos County, which has population 34489 and is centered at ( -105.6388781 , +36.5771832 )
Node 22 is Grant County, which has population 28185 and is centered at ( -108.3815043 , +32.7320870 )
Node 23 is Colfax County, which has population 12387 and is centered at ( -104.6401105 , +36.6129638 )
```

In [4]:
```
# Geodesic can be used to find the distance between x and y coordinates
# A test of this for counties Mora and De Baca are shown below

from geopy.distance import geodesic

Mora = ( G.nodes[31]['C_Y'], G.nodes[31]['C_X'] )
De_Baca = ( G.nodes[16]['C_Y'], G.nodes[16]['C_X'] )

print( "Mora -> De Baca County:", geodesic(Mora, De_Baca).miles )
```

```
Mora -> De Baca County: 116.22165034291885
```

In [5]:
```
# create distance dictionary for the distance between any counties i and j

dist = dict()
for i in G.nodes:
    for j in G.nodes:
        loc_i = ( G.nodes[i]['C_Y'],  G.nodes[i]['C_X'] )
        loc_j = ( G.nodes[j]['C_Y'],  G.nodes[j]['C_X'] )
        dist[i,j] = geodesic(loc_i,loc_j).miles
```

In [6]:
```
# Let's impose a 1% population deviation (+/- 0.5%)
deviation = 0.01

import math

k = 3          # number of districts
total_population = sum(G.nodes[node]['TOTPOP'] for node in G.nodes)  # Population of New Mexico

L = math.ceil((1-deviation/2)*total_population/k) # Lower Bound of District Population
U = math.floor((1+deviation/2)*total_population/k) # Upper Bound of District Population

print("Using L =",L,"and U =",U,"and k =",k)
```

```
Using L = 702312 and U = 709369 and k = 3
```

In [7]:
```
import gurobipy as gp
from gurobipy import GRB

# create model
m = gp.Model()

# create x[i,j] variable which equals one when county i is assigned to (district centered at) county j
x = m.addVars(G.nodes, G.nodes, vtype=GRB.BINARY)
```

```
Restricted license - for non-production use only - expires 2024-10-28
```

In [8]:
```
# objective is to minimize the moment of inertia: d^2 * p * x
m.setObjective( gp.quicksum( dist[i,j]*dist[i,j]*G.nodes[i]['TOTPOP']*x[i,j] for i in G.nodes for j in G.nodes), GRB.MINIMIZE )
```

In [9]:
```
# add constraints saying that each county i is assigned to one district
m.addConstrs( gp.quicksum(x[i,j] for j in G.nodes) == 1 for i in G.nodes)

# add constraint saying there should be k district centers
m.addConstr( gp.quicksum( x[j,j] for j in G.nodes ) == k )

# add constraints that say: if j roots a district, then its population is between L and U.
m.addConstrs( gp.quicksum( G.nodes[i]['TOTPOP'] * x[i,j] for i in G.nodes) >= L * x[j,j] for j in G.nodes )
m.addConstrs( gp.quicksum( G.nodes[i]['TOTPOP'] * x[i,j] for i in G.nodes) <= U * x[j,j] for j in G.nodes )

# add coupling constraints saying that if i is assigned to j, then j is a center.
m.addConstrs( x[i,j] <= x[j,j] for i in G.nodes for j in G.nodes )

m.update()
```

In [10]:
```
# Add contiguity constraints

import networkx as nx
DG = nx.DiGraph(G)

# Add variable f[j,u,v] which equals the amount of flow (originally from j) that is sent across arc (u,v)
f = m.addVars( DG.nodes, DG.edges, vtype=GRB.CONTINUOUS)
M = DG.number_of_nodes()-1

# Add constraint saying that node j cannot receive flow of its own type
m.addConstrs( gp.quicksum( f[j,u,j] for u in DG.neighbors(j) ) == 0 for j in DG.nodes )

# Add constraints saying that node i can receive flow of type j only if i is assigned to j
m.addConstrs( gp.quicksum( f[j,u,i] for u in DG.neighbors(i)) <= M * x[i,j] for i in DG.nodes for j in DG.nodes if i != j )

# If i is assigned to j, then i should consume one unit of j flow.
# Otherwise, i should consume no units of j flow.
m.addConstrs( gp.quicksum( f[j,u,i] - f[j,i,u] for u in DG.neighbors(i)) == x[i,j] for i in DG.nodes for j in DG.nodes if i != j )

m.update()
```

```
Cutting planes:
  Cover: 3
  MIR: 3
  StrongCG: 2
  Flow cover: 7
  Network: 2
  Relax-and-lift: 2

Explored 1 nodes (772 simplex iterations) in 0.54 seconds (0.23 work units)
Thread count was 8 (of 8 available processors)

Solution count 3: 1.26547e+10 1.27093e+10 1.27927e+10

Optimal solution found (tolerance 0.00e+00)
Best objective 1.265473701570e+10, best bound 1.265473701570e+10, gap 0.0000%
```

In [12]:
```
print("The moment of inertia objective is",m.objval)

# retrieve the districts and their populations
centers = [j for j in G.nodes if x[j,j].x > 0.5 ]
districts = [ [i for i in G.nodes if x[i,j].x > 0.5] for j in centers]
district_counties = [ [ G.nodes[i]["NAME20"] for i in districts[j] ] for j in range(k)]
district_populations = [ sum(G.nodes[i]["TOTPOP"] for i in districts[j]) for j in range(k) ]

# print district info
for j in range(k):
    print("District",j,"has population",district_populations[j],"and contains counties",district_counties[j])

# What is the population deviation for this model?
population_deviation = max(district_populations) - min(district_populations)
print("\nThe population deviation for this model is", population_deviation, "people.")
```

```
The moment of inertia objective is 12654737015.702263
District 0 has population 707195 and contains counties ['Bernalillo', 'Catron', 'Cibola']
District 1 has population 705274 and contains counties ['Curry', 'Roosevelt', 'Lincoln', 'Guadalupe', 'Luna', 'Lea', 'Chaves', 'Quay', 'Sierra', 'De Baca', 'Otero', 'San Miguel', 'Grant', 'Doña Ana', 'Hidalgo', 'Socorr
o', 'Eddy']
District 2 has population 705053 and contains counties ['San Juan', 'Torrance', 'McKinley', 'Rio Arriba', 'Santa Fe', 'Valencia', 'Union', 'Taos', 'Colfax', 'Sandoval', 'Harding', 'Los Alamos', 'Mora']

The population deviation for this model is 2142 people.
```

- Read County Graph: Read the New Mexico county graph from a JSON file using the 'gerrychain' library.
- Retrieve County Information: Iterate through nodes to retrieve county information, including name, population, longitude, and latitude. Print the information for each county.
- Calculate Distances: Use the 'geopy' library to calculate distances between the centers of two counties (Mora and De Baca in this case).
  Define Optimization Parameters:
  Set the population deviation, number of districts (k), and calculate lower and upper bounds for district populations.
- Gurobi Model Setup: Create a Gurobi optimization model (m).
  Add binary decision variables (x) to represent county assignments to district centers.
  Set the objective function to minimize the moment of inertia, considering distances and populations.
- Add Constraints: Ensure each county is assigned to exactly one district. Enforce that there are exactly k district centers.
  Set constraints on district populations based on the specified bounds.
  Introduce coupling constraints to relate assignment variables.
- Contiguity Constraints: Add flow variables (f) to model the flow of districts in the directed graph. Introduce constraints to ensure contiguity and proper flow within assigned districts.
- Solve the Model: Optimize the Gurobi model to find the optimal solution. Print the objective value, representing the moment of inertia.
- Retrieve District Information: Identify district centers and the counties assigned to each district.Calculate district populations.
- Population Deviation: Calculate the population deviation among districts.
- Create Map: Use the 'geopandas' library to read a shapefile of New Mexico counties. Assign counties to districts based on the optimization results. Create a map displaying the district assignments.
- Results Summary: Print information about each district, including population and counties. Print the population deviation. Display the map illustrating the district assignments.

The code demonstrates the formulation and solution of a spatial optimization problem related to districting, considering both population constraints and geographical contiguity.

Experiments:

Both models underwent computational experiments on a 2021 Windows Surface Laptop 3, featuring the following specifications:

Operating System: Microsoft Windows 10 Pro

System Type: 64-bit operating system, x64-based processor

Processor Speed: 1.30 GHz

RAM: 16.0 GB

The optimization solver employed for both models was Gurobi Optimizer Version 10.0.0, build v10.0.0rc2 (win64). The implementation was carried out using Python, and the code execution took place using Jupyter notebook.

Minimize Cut Edges (Model 1):

The optimization results for Model 1 were as follows:

Objective Value of Optimization Model: 19 cut edges

Time Required: 0.7 seconds

Minimize Moment of Inertia (Model 2):

The optimization results for Model 2 were as follows:

Objective Value of Optimization Model: 12,654,737,015.702263 (people•mi^2)

Time Required: 0.6 seconds

These experiments provide insights into the computational efficiency of both models, showcasing the objective values achieved and the time required for optimization on the specified hardware and software environment. The performance metrics serve as valuable benchmarks for evaluating the practical feasibility and effectiveness of the redistricting models.

Plans and Maps:

- **Minimize Cut Edges (Model 1) Plan:**

District 1:

Counties: Curry, San Juan, Guadalupe, McKinley, Rio Arriba, Santa Fe, Quay, Union, San Miguel, Taos, Colfax, Sandoval, Harding, Los Alamos, Mora

Population: 702,632 people

District 2:

Counties: Bernalillo, Catron, Cibola

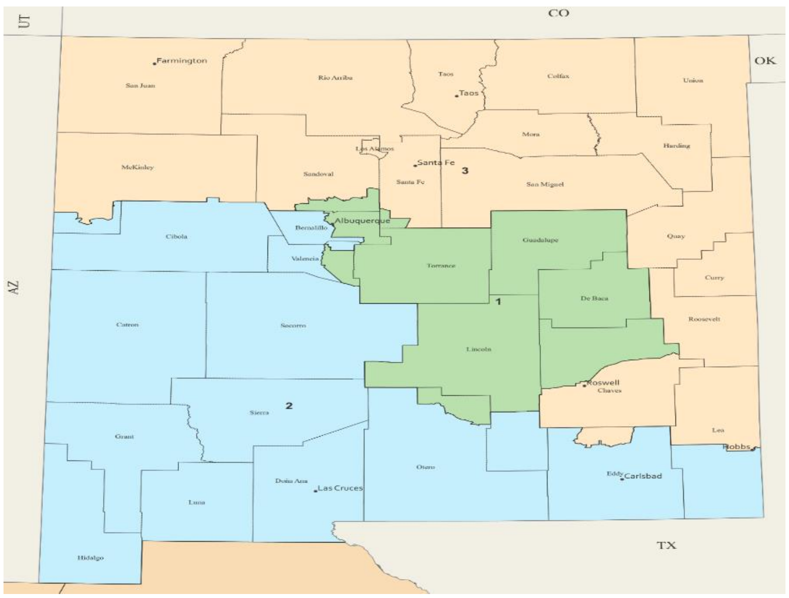
Population: 707,195 people

District 3:

Counties: Roosevelt, Lincoln, Torrance, Luna, Lea, Chaves, Sierra, Valencia, De Baca, Otero, Grant, Doña Ana, Hidalgo, Socorro, Eddy

Population: 707,695 people

Total Population Deviation: 5,063 people (0.239%)



Model 1 Map

- **Minimize Moment of Inertia (Model 2) Plan:**

District 1:

Counties: Bernalillo, Catron, Cibola

Population: 707,195 people

District 2:

Counties: Curry, Roosevelt, Lincoln, Guadalupe, Luna, Lea, Chaves, Quay, Sierra, De Baca, Otero, San Miguel, Grant, Doña Ana, Hidalgo, Socorro, Eddy

Population: 705,274 people

District 3:

Counties: San Juan, Torrance, McKinley, Rio Arriba, Santa Fe, Valencia, Union, Taos, Colfax, Sandoval, Harding, Los Alamos, Mora

Population: 705,053 people

Total Population Deviation: 2,142 people (0.101%)

This alternative presentation offers a concise breakdown of each district's composition and population, providing a clearer view of the redistricting plan under both Model 1 and Model 2.

Evaluation of Plans:

To meet all criteria, a redistricting plan must adhere to the minority representation rule in addition to other specified conditions. In New Mexico, the two significant minority populations are Hispanics and Native Americans, constituting 47.5% and 10.9% of the total population, respectively. Ensuring the integrity of these populations is crucial for fair representation.

The Hispanic population, approximately 989,000, is concentrated in counties like Bernalillo and Doña Ana, with populations of 328,000 and 143,000, respectively. Given their geographic dispersion, it's challenging to keep these counties contiguous in a single district. However, preserving the entirety of counties contributes to maintaining minority representation across separate regions.

Both presented redistricting plans fulfill all criteria, including adherence to federal regulations like the Voting Rights Act of 1965. The first model prioritizes maximizing district compactness while keeping counties whole, resulting in a population deviation of 0.293%, or 5,063 people. The second model achieves a smaller population deviation of 0.101%, or 2,142 people, making it more favorable.

Comparatively, the second model exhibits less political bias than the first. Political affiliation maps reveal that the first model's selection would lead to districts dominated entirely by one party, limiting political variation. In contrast, the second model offers a more balanced representation of political diversity across districts.

This alternative presentation underscores the significance of minority representation, provides clarity on population deviations, and emphasizes the political neutrality of the redistricting plans.

Conclusions:

After a meticulous examination of both proposed redistricting plans, it is evident that each plan meticulously aligns with the specified federal and state criteria. The second plan, grounded in the concept of moment of inertia, emerges as the preferred choice for future congressional districting.

The recommended plan strategically divides the counties into three districts, boasting populations of 707,195, 705,274, and 705,053 people, with an impressively low total population deviation of 0.101%, equivalent to 2,142 people. This model excels in key aspects such as minimized population deviation, enhanced compactness, contiguous county divisions, preservation of minority interests, and political diversity.

The emphasis on these crucial factors positions the second model as a robust candidate for adoption in the redistricting process. The plan's commitment to fairness, equitable representation, and inclusivity aligns with the principles underlying the democratic process. If implemented, this redistricting plan has the potential to empower citizens in New Mexico, ensuring their voices are accurately reflected in the nation's future.

Respiratory

https://davesredistricting.org/maps#viewmap::ec1c76cd-f59f-445b-8f24-fbffb0e8bdf5

References:

Lee, Morgan. “New Mexico governor approves 3-district congressional map.” Las Cruces Sun News, https://www.lcsun-news.com/story/news/local/new-mexico/2021/12/17/new mexico-governor-approves-3-district-congressional-map/8942109002/. Accessed 23 Mar. 2023.

Buchanan, Austin L. “Districting Examples using Python, Gurobi, NetworkX, and GeoPandas.” GitHub, https://github.com/AustinLBuchanan/Districting-Examples, Accessed 29, Apr. 2023

“History.” New Mexico Indian Affairs Department. www.iad.state.nm.us/about-us/history/#:~:text=New%20Mexico. Accessed 7 May 2023

“The Demographic Statistical Atlas of the United States - Statistical Atlas.” Statisticalatlas.com, statisticalatlas.com/state/New-Mexico/Race-and-Ethnicity. Accessed 2 May 2023.

New Mexico Legislative Council Service. “A Guide to State and Congressional Redistricting.” 2011. Accessed 26 Apr. 2023.

US Census Bureau. “Our Censuses.” Census.Gov, https://www.census.gov/programs surveys/censuses.html. Accessed 26 Apr. 2023

Dra 2020. Daves Redistricting. (n.d.).

https://davesredistricting.org/maps#viewmap::ec1c76cd-f59f-445b-8f24-fbffb0e8bdf5