Preliminary Master Use Case List (5/7/24)

CSE416, S01 – Spring 2024

This draft of the use case list is meant to help you think about requirements. You should think about additional use cases that serve the project goals and also think about more detail for each use case.

Project goals

Perform racial/ethnic analysis of political data, specifically:

- Determine the racial/ethnic distribution of the current state assembly,
- Analyze the distribution of racial/ethnic groups within states,
- Determine if district boundaries in states disadvantage certain groups,
- Analyze in-state voting behavior of racial/ethnic groups,
- Compare results of one state with a history of voter discrimination with one state with no such (or limited) discrimination history, and
- Identify racial/ethnic Gerrymandering.

Terminology

- Population – the population of a region (e.g., state, precinct, etc.) refers to the total population as defined by the US Census Bureau. Some calculations might refer to the voting age population (VAP) or citizen voting age population (CVAP). You can use any measure of population, but it should be done consistently throughout the application.
- Ideal district population is the total population of the state divided by the number of districts in the state.
- Ensemble – a collection of district plans generated on the SeaWulf. Each such district plan will be random and will be a subset of all of the possible graph partitions that are constrained by the limits on population equality.

Notation

Use cases listed below include a categorization following the use case title. Three categories of use cases are provided as "required," "preferred," and "optional." Use cases with an "SD," "AD," or "part of GUI SD" indicate that the use case might be requested in the design review with "SD" referring to a sequence diagram and "AD" referring to an activity diagram. For some use cases, the GUI and the server part can be combined into one use case diagram. Many of the use cases that are related and that require activity diagrams can and should be combined into one activity diagram.

Some of the use cases are closely related and might be combined into a single design document. The reason for closely related use cases is that we use our use cases as units of work, so there is an attempt to provide additional use cases for some more difficult tasks.

Ensemble sizes

For testing purposes, you will need to create different size ensembles. Your maximum size ensemble should include 5,000 plans for each state.

General GUI (12 required)

GUI-1. Select state to display (required) (SD)

The user can pick a state through a dropdown menu or possibly through clicking on the state in a map of the US. The state selection will cause a table to be displayed that contains a summary of the ensembles available for the state. Ensemble data in the table will include the number of district plans in the ensemble and the population equality threshold used in the MCMC computation. In addition, state selection will also cause the map of the state to be displayed as described in GUI-2.

GUI-2. Display the current district plan when state is selected (required) (SD)

After selecting a state either from the map or the dropdown, by default, the user should be shown the current (2022) State Assembly district plan displayed on the centered state map at a zoom level appropriate to the size and location of the state.

GUI-3. State data summary (required) (SD)

The data associated with the state will be summarized in response to the user selecting the state and shown concurrently with the map of the state (GUI-2). At a minimum, the summary data will include state population, state voter distribution (estimate based on a recent statewide election), state total population, population of each significant racial/ethnic group in the state, party control of the redistricting process (if any), and summary of state representatives by party and by racial/ethnic group.

GUI-4. Display demographic heat map by precinct or district (required) (SD)

When the user selects a minority group from a drop-down menu, a heat map for the demographic group in the state will be displayed. The monochromatic heat map will show the percentage of the selected group in each precinct. Choose a number of bins that effectively shows the population distribution with bin ranges that are equal use bounds that are integer values of population percentage. The map will include a legend that displays the bin ranges and associated colors.

GUI-5. Display demographic heat map by census block or precinct (preferred)

When the user selects a minority group from a drop-down menu, a heat map for the demographic group in the state will be displayed. The monochromatic heat map will show the percentage of the selected group in each census block or county. Choose a number of bins that effectively shows the population distribution with bin ranges that are equal use bounds that are integer values of population percentage. The map will include a legend that displays the bin ranges and associated colors. Precinct option cannot be taken for both this use case and the previous.

GUI-6. Display state assembly table (required) (SD)

When the user clicks on screen component selecting district detail (or some other appropriate trigger), a table will be displayed. Each row in the table will contain data for one state assembly district. At a minimum, the data will contain the district number, the representative, the representative's party, the representative's racial/ethnic group, and the vote margin as a percentage in the most recent election.

GUI-7. Display photo of district representative (preferred) (SD)

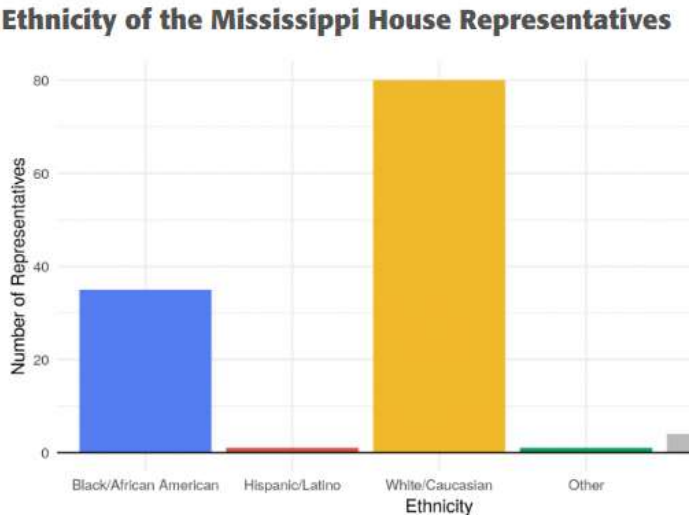
A recent photo of each district representative will be displayed, either directly in the table (GUI-5), or by clicking on a component in the table, or by hovering over a component in the table.

GUI-8. Display district plan (required)

If a user clicks on some identifier of a district in the state assembly table, the district will be highlighted on the map. Highlighting can be performed in a variety of ways. For example, the border of the highlighted district might change color or thickness.

GUI-9. Display racial/ethnic distribution of current Assembly (required)

Display a bar chart that compares the racial/ethnic background of current members of the state Assembly. All groups that have the possibility of electing a representative of their choice should be included in the chart. The sample presented in class is shown below.



GUI-10. Display a table of feasible opportunity districts (preferred)

Display a table that contains a row for each possible racial/ethnic group. Each row contains the ideal population of an assembly district, the population of that minority, actual number of opportunity districts in the current plan, and your estimate of the maximum number of opportunity districts, and the average number of opportunity districts in your ensemble. Calculate the maximum number of opportunity districts as a multiple of 50% of the ideal district population. For example, if the minority population was 100,000 and the ideal district population was 20,000, the maximum number of opportunity districts would be 10.

GUI-11. Display one or more district-sized opportunity districts (preferred)

When clicking on a link in row in the table specified above, the borders of the opportunity districts in the current plan are displayed in the map of the state. The border of each such opportunity district will have a color different from the color for the other districts.

GUI-12. Reset page (preferred)

When the user clicks a reset button, the GUI will reset to the condition before the user selected a state.

GUI-13. Compare two district plans on the map (preferred) (SD)

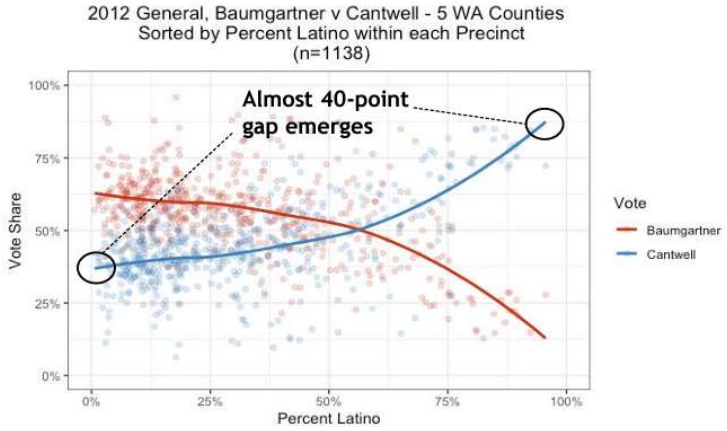
Compare two district plans by showing both plans on the map and displaying a tabular comparison of the distance measures for each plan. This could be limited to comparing a selected random plan with the enacted plan. The trigger will be some GUI component (e.g., "Compare with enacted" button).

GUI-14. Display a bar chart of opportunity districts in the ensemble (required)

When a user selects this option (along with a specified racial/ethnic group), the GUI will display a bar chart showing the distribution of opportunity districts in the ensemble.

GUI-15. Display Gingles 2/3 analysis results (required)

In response to a user request, display a scatter plot (example below) that shows the precinct election results for each party organized on an x, y axis by percentage or racial/ethnic group in the precinct and party vote share. Any of the feasible racial/ethnic groups in the state should be capable of being displayed.



GUI-16. Display the Gingles 2/3 analysis result for a second election (preferred)

Use a second election to display the results of the Gingles 2/3 analysis.

GUI-17. Display the Gingles 2/3 analysis data in a tabular display (preferred)

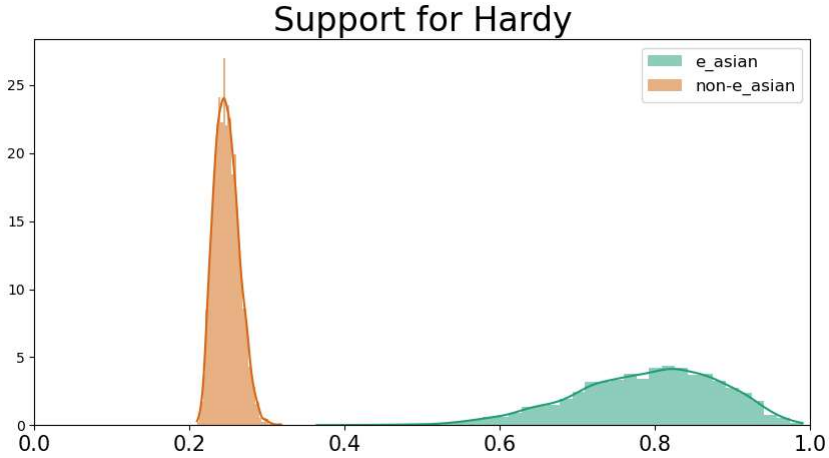
In response to a user request, a table display of the precinct-by-precinct results will be displayed. Each row will display the data for a precinct including, total population, minority population, Republican votes, and Democratic votes.

GUI-18. Highlight a Gingles 2/3 table row (preferred)

In response to a user selecting a dot in the Gingles scatter plot, the precinct identified by the dot will be highlighted in the table above.

GUI-19. Display candidate results of Ecological Inference (EI) analysis (required)

Display the results of the EI analysis in response to a user GUI request. The user shall have the ability to select the election and the racial/ethnic groups to compare. The results will be shown in a display for each candidate (example below) in which the x-axis represents the percentage of a racial/ethnic group that voted for a candidate and the y-axis represents the associated probability value for each x-axis value.



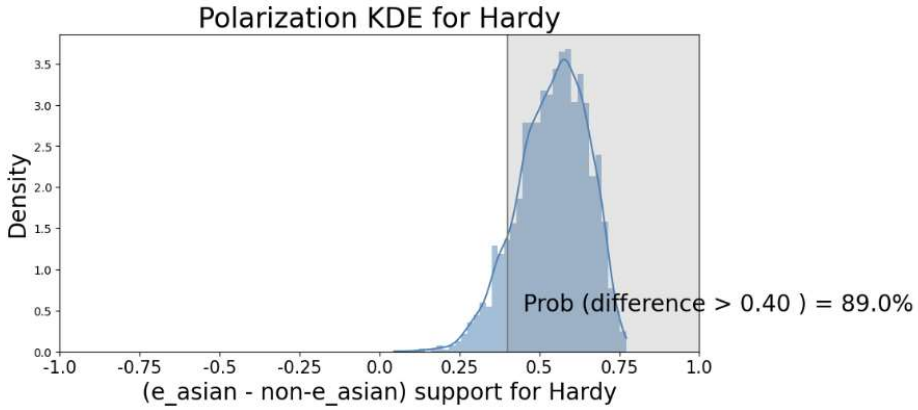
GUI-20. Display EI precinct results in a choropleth maps (preferred)

Display the EI results in a choropleth maps (one map per candidate) in which each precinct is displayed in a color that is consistent with the most likely level of support for the candidate in the precinct.

GUI-21. This use case maintains ordering of GUI use cases

GUI-22. Display EI KDE results (preferred)

Display the EI KDE results that compare support for a candidate between two racial/ethnic groups (e.g., white and African American).



GUI-23. Display box & whisker data (required) (SD)

The user will be able to request the display of box & whisker data for an ensemble of district plans and a particular racial/ethnic group from among the feasible groups in the state. The display will show the results of the ensemble box & whisker analysis for each group. Dots for each district in the current enacted district plan will be shown in the display (in order of increasing percentage of the minority group). The display should be sufficient in size to show your largest state and should include a legend and color selection to make the chart easily readable.

GUI-24. Display vote share vs seat share curve (preferred)

If your Gingles-2/3 test shows racially polarized voting, display the vote share vs seat share curve for the state. If the Gingles-2/3 test does not indicate racially polarized voting in one of your states, the GUI component that allows the user to select the display should be disable.

GUI-25. Filter Representative data (optional)

On the state assembly table, the user will have the ability to filter the data by ethnic group, party, and state. The user can do so through a drop-down menu for each option located in a nearby component. The user can select a choice in the dropdown menu, which in turn causes the table to reflect the choice.

Preprocessing (7 required)

Prepro-1. Integrate multiple data sources (required) (AD)

Integrate and merge US Census data (population, both for total and for the any opportunity groups), precinct data (boundary, name, demographics, etc.), and existing district data (boundary, name, district#, etc.). Geographic boundary data should be converted (if necessary) to a consistent format (e.g., GeoJSON).

Prepro-2. Identify precinct neighbors (required) (AD)

Identify two precincts as neighbors if they share a common boundary of at least 200 feet and the edges of each precinct are within 200 feet of its neighbors' edges. If possible, try to locate a data source for which this computation is already done.

Prepro-3. Integrate enacted plan with dataset (required)

Integrate the enacted plan for the state within the server database.

Prepro-4. Store preprocessed data (required)

The preprocessed data should be stored in the NoSQL or relational database. If a relational database is used, the data should be stored in third normal form. Data might also be stored in a file system accessible to the server.

Prepro-5. Store SeaWulf data (required)

Retrieve generated data from SeaWulf for each of your states, convert to an appropriate format, and store either in your database or in a file system. Data stored in a file system should be accessible through a path obtained from your database.

Prepro-6. Calculate statewide measures (required)

Overall state measures include percentage of Republican voters, percentage of Democratic voters, percentage of each of the demographic groups relevant for the state. A demographic group is considered relevant for a state if the population of the group is sufficient to provide a winning vote margin in at least one district. You should estimate the percentage of voters in each party by analyzing the votes for each party in each precinct in some statewide election (e.g., presidential, attorney general, governor).

Prepro-7. Generate data files required for SeaWulf processing (required) (AD)

Generate all the data files required for SeaWulf processing. This will include the graph representation of the precincts in a state as well as geographic, election, and incumbent data for each precinct.

Prepro-8. Determine the racial/ethnic distribution of the state Assembly (required)

Determine the racial/ethnic background of each of the representatives to the State Assembly for your two states. Use public biographical states and other sources as your basis. If a representative were included in multiple groups, select the one group that appears most prominent.

Prepro-9. Determine possible opportunity districts (preferred)

For each of the feasible racial/ethnic groups (population of the group is more than 50% of that of the ideal Assembly district) in each state, determine if a contiguous district could be constructed.

Prepro-10. Gingles 2/3 precinct analysis (required)

Perform a precinct-by-precinct analysis of voting results and minority population percentage for some statewide race (2022 or 2020). For each precinct, the analysis will identify the winning party, the Republican vote share, the Democratic vote share, and the population percentage of the racial/ethic group. The analysis is repeated for each feasible racial/ethnic group in the state.

Prepro-11. Gingles 2/3 non-linear regression analysis (required)

For the statewide race used in the use case above, calculate the non-linear regression curve for the Republican and Democratic precinct values for each feasible racial/ethnic group in the state. For each of the curves, the x, y values are the population percentage of the racial/ethnic group and the vote share for each party. Multiple equation forms will be used to determine the best form for the non-linear regression.

Prepro-12. Use the PyEI MGGG software to calculate Ecological Inference data (required)

Use the PyEI MGGG software to calculate results for multiple statewide races.

Prepro-13. Calculate the vote share vs seat share curve data (preferred)

Using the Shen software as a starting point, calculate the data for the vote share vs. seat share curve in any of your states that displays racially polarized voting. Use the current district plan as the basis for the calculation. Also use relatively fine grain increments of vote share and possibly randomization to reduce a stair-stepping effect.

Prepro-14. Determine racial distribution of the State Assembly algorithmically (optional)

Determine racial distribution of the State Assembly algorithmically. Use the DeepFace model to classify the races of state representatives by analyzing their images. The model predicts potential races with confidence scores, and you select the race with the highest confidence value for each representative. Be sure to cross-check the origins of the representative's name as predictions are inaccurate in some cases, due to lighting of images, bias in model, etc.

SeaWulf (16 required)

SeaWulf-1. Server dispatcher (required)

Establish a protected directory on SeaWulf to store your team's data. Pre-stage any data that might be used repeatedly for SeaWulf runs. Prior to submitting a batch districting run request to SeaWulf, the data required for the run should be marshalled (from memory and/or DB), and passed to the SeaWulf as a file (or multiple files) to be stored in the team's SeaWulf file system.

SeaWulf-2. Run MGGG ReCom algorithm on the SeaWulf (required) (AD)

Set the constants in the MGGG code to define the properties (e.g., constraints) of the run. Any run-control information should be packaged in a SeaWulf acceptable format (e.g., script commands) and executed on SeaWulf. Your activity diagram should demonstrate that you understand how the MGGG algorithm operates. You should generate at least 2 ensembles for each state. The small ensemble will contain approximately 250 random district plans and the larger ensemble will contain approximately 5,000 plans.

SeaWulf-3. Coordinate/aggregate SeaWulf core generated data (required) (AD)

You will run your code on a single SeaWulf node, one that has multiple cores. Each of the cores will generate one or more random graph partitions (i.e., district plans) and store a concise version of those results in a shared file directory for your team. Following the completion of each random graph partition, the core begins the generation of the next random district plan. You will coordinate the work of the multiple cores so that when the target number of district plans is completed, each of the cores ends its processing.

SeaWulf-4. Calculate election winners (required) (AD)

Using 2022 statewide election results, estimate the election results in each district of each ensemble district plan. You will calculate this by summing up the estimated votes in each node (i.e., precinct) of a partition sub-graph. You can use a suitable precinct by precinct vote in a 2022 statewide election. If there is no 2022 statewide election, you can use the 2020 presidential election results.

SeaWulf-5. Calculate the Republican/Democratic split for each random district plan (required) (AD)

For each generated plan in an ensemble, estimate the Republican/Democratic votes in each district. Since each district is a collection of precincts, use the historic precinct vote totals (e.g., 2020 Presidential, 2022 Gubernatorial, etc.) to estimate the winner of an election in each district.

SeaWulf-6. Identify and store additional random district plans of note (required) (AD)

You will not be able to store all of your random district plans in the server database, but you will store some subset of those plans. Summary information and detailed information of such plans should be stored in your server database for eventual display by the user. A district plan of note includes the plans with the maximum and minimum number of opportunity districts for each of the racial/language groups.

SeaWulf-7. Calculate ensemble measures (required) (AD)

Calculate the summary measures for each ensemble. At a minimum, measures will include the number of district plans, the number of opportunity districts for each significant minority, and Republican/Democratic splits.

SeaWulf-8. Run on multiple SeaWulf nodes (preferred)

Run the MGGG algorithm and supplemental code on multiple nodes on SeaWulf. Node coordination should be done using a suitable coordination mechanism (e.g., MPI). Be able to estimate speed-up during your final project presentation. Note that this is optional since your algorithms should run effectively on a large-core single node processor.

SeaWulf-9. Python profiler (preferred)

Profile your system performance on SeaWulf using a Python profiler tool. Identify the procedures that consume the most CPU time. Results can be displayed using some Python-appropriate tool and displayed as an image in your final presentation.

SeaWulf-10. Identify opportunity districts in each random district plan (required)

For each district plan, identify all opportunity districts. Use at least three thresholds (percentage of the given racial/ethnic group in the district) to determine the districts, one of which is 37% and another one is 50%.

SeaWulf-11. Identify "interesting" random district plans (preferred)

Identify "interesting" district plans that will be available for display in the GUI. Examples of such plans are those with a maximum number of opportunity districts for each racial/ethnic group. Another category would be random plans with an average number of opportunity districts for each racial/ethnic group.

SeaWulf-12. Calculate box & whisker data (required) (AD)

Calculate the box & whisker summary data for all the random district plans generated by the SeaWulf. These calculations will made be for each feasible racial/ethnic group in the state.

Server Processing (0 required)

All server processing addressed in GUI use cases.

Additional Use Cases