# Course project: stage 1

PSTAT131-231

**Project overview and expectations**

Your final project will be to merge census data with 2016 voting data to analyze the election outcome. The work will be carried out in two stages:

1. Preparation and planning (guided)

- Background reading
- Data preparation
- Exploratory analysis
- Tentative plan for statistical modeling

2. Data analysis and reporting (open-ended)

- Statistical modeling
- Interpretation of results
- Report findings

This document pertains to the first stage: you'll gather background, preprocess and explore the data, and come up with a tentative plan for the second stage.

Your objective is to work through the steps outlined in this document, which walk you through data preparation and exploration. The structure is similar to a homework assignment, and your deliverable will be a knitted PDF with all steps filled in.

**Formatting guidelines**

- Your knitted document should not include codes.
- All R output should be nicely formatted; plots should be appropriately labeled and sized, and tables should be passed through `pander()`. Raw R output should not be included.
- Avoid displaying extra plots and figures if they don't show information essential to addressing questions.

**Suggestions for teamwork**

- Set a communication plan – how will you share your work and when/how will you meet?
- Assign roles – designate a group member to coordinate communication and another group member to coordinate preparation and submission of your deliverables.
- Divide the work! Discuss your skills and interests and assign each group member specific tasks. Many of the tasks can be carried out in parallel. For those that can't, if some of your group members have more immediate availability, have them work on earlier parts, and have other members follow up on their work by completing later parts.

**Other comments**

- The plan that you lay out at the end of this document is not a firm committment – you can always shift directions as you get farther along in the project.
- Negative results are okay. Sometimes an analysis doesn't pan out; predictions aren't good, or inference doesn't identify any significant associations or interesting patterns. Please don't feel that the tasks you propose in this first stage need to generate insights; their merit will be assessed not on their outcome but on whether they aim at thoughtful and interesting questions with a reasonable approach.

**Evaluations**

Our main objective at this stage is to position you well to move forward with an analysis of your choosing, and to provide feedback on your proposal. We may suggest course corrections if we spot anything that we anticipate may pose significant challenges downstream, or encourage you to focus in a particular direction when you start your analysis. Our goal is *not* to judge or criticize your ideas, but rather to help make your project a more rewarding experience. Most credit will be tied to simply completing the guided portions. Here are the basic criteria.

- We'll look for the following in the guided portions (Part 0 – Part 2):
  - Has your group completed each step successfully?
  - Does your document adhere to the formatting guidelines above?
- We'll look for the following in your proposed tasks:
  - Is the task relevant to understanding or predicting the election outcome?
  - Is a clear plan identified for how to prepare the data for statistical modeling that is appropriate for the task?
  - Is the modeling approach sensible given the task?

# Part 0. Background

The U.S. presidential election in 2012 did not come as a surprise. Some correctly predicted the outcome of the election correctly including Nate Silver, and many speculated about his approach.

Despite the success in 2012, the 2016 presidential election came as a big surprise to many, and it underscored that predicting voter behavior is complicated for many reasons despite the tremendous effort in collecting, analyzing, and understanding many available datasets.

To familiarize yourself with the general problem of predicting election outcomes, read the articles linked above and answer the following questions. Limit your responses to one short paragraph (3-5 sentences) each.

**Question 0 (a)**

What makes voter behavior prediction (and thus election forecasting) a hard problem?

Although we are interested in how people will vote on election day, the data we get is based on how people think they will vote at the time they are asked, that changes over time. This change might be because of something measurable, such as a change in employment, or if the president doubles federal income tax or can be changes that we can't measure, such as an impactful campaign ad. Another source of variation is sampling error where voters of one candidate is overrepresented in the sample, this error can be estimated to some degree and adjusted for. Also if polls are conducted over phone or email there is the case of non-response and/or lying about their voting preference, the corrections by pollsters may be biased due to "house effect".

**Question 0 (b)**

What was unique to Nate Silver's approach in 2012 that allowed him to achieve good predictions?

The mathematical model for the proportion of people saying they will vote for a candidate is the actual percentage + "house effect" + sampling variation. While 'house effect' and sampling variation can be estimated to a high degree based on past polling data, the actual percentage can have temporal shift based on various socio-political-economic factors. Every polling survey done over time provides a time series data of polling percentages. The traditional method was to compute the maximum of the polling percentages which are $> 50\%$, Nate Silver computed the number of time series which end up above the 50% mark among all time series available. For states where no polling data was available, hierarchical clustering and Graph theory was used to estimate their polls.

**Question 0 (c)**

What went wrong in 2016? What do you think should be done to make future predictions better?

Polls and forecasts were generally wrong about the election due to systematic polling challenges. Clinton's projected voteshare was overestimated in most cases, particularly within swing states. Experts have speculated that Trump supporters were reluctant to participate and/or answer honestly to polling questions. In the future, I think pollsters should rely less on traditional landline calls and incorporate a wider variety of information-gathering methods, such as cellphone calls, emails, texts, internet polls, and social media web-scraping.

# Part 1. Datasets

The `project_data.RData` binary file contains three datasets: tract-level 2010 census data, stored as `census`; metadata `census_meta` with variable descriptions and types; and county-level vote tallies from the 2016 election, stored as `election_raw`.

## Election data

Some example rows of the election data are shown below:

| county | fips | candidate | state | votes |
|---|---|---|---|---|
| Los Angeles County | 6037 | Hillary Clinton | CA | 2464364 |
| Los Angeles County | 6037 | Donald Trump | CA | 769743 |
| Los Angeles County | 6037 | Gary Johnson | CA | 88968 |
| Los Angeles County | 6037 | Jill Stein | CA | 76465 |
| Los Angeles County | 6037 | Gloria La Riva | CA | 21993 |
| Cook County | 17031 | Hillary Clinton | IL | 1611946 |

The meaning of each column in `election_raw` is self-evident except `fips`. The accronym is short for Federal Information Processing Standard. In this dataset, `fips` values denote the area (nationwide, statewide, or countywide) that each row of data represent.

Nationwide and statewide tallies are included as rows in `election_raw` with `county` values of `NA`. There are two kinds of these summary rows:

- Federal-level summary rows have a `fips` value of `US`.
- State-level summary rows have the state name as the `fips` value.

### Question 1 (a)

Inspect rows with `fips == 2000`. Provide a reason for excluding them.

Alaska only has one county, so the fips county data is a duplicate of the statewide data.

### Question 1 (b)

Drop these observations – please write over `election_raw` – and report the data dimensions after removal.

| rows | columns |
|---|---|
| 18345 | 5 |

## Census data

The first few rows and columns of the `census` data are shown below.

| CensusTract | State | County | TotalPop | Men | Women |
|---|---|---|---|---|---|
| 1001020100 | Alabama | Autauga | 1948 | 940 | 1008 |
| 1001020200 | Alabama | Autauga | 2156 | 1059 | 1097 |
| 1001020300 | Alabama | Autauga | 2968 | 1364 | 1604 |
| 1001020400 | Alabama | Autauga | 4423 | 2172 | 2251 |
| 1001020500 | Alabama | Autauga | 10763 | 4922 | 5841 |
| 1001020600 | Alabama | Autauga | 3851 | 1787 | 2064 |

Variable descriptions are given in the `metadata` file. The variables shown above are:

| variable | description | type |
|---|---|---|
| CensusTract | Census tract ID | numeric |
| State | State, DC, or Puerto Rico | string |
| County | County or county equivalent | string |
| TotalPop | Total population | numeric |
| Men | Number of men | numeric |
| Women | Number of women | numeric |

## Data preprocessing

### Election data

Currently, the election dataframe is a concatenation of observations (rows) on three kinds of observational units: the country (one observation per candidate); the states (fifty-ish observations per candidate); and counties (most observations in the data frame). These are distinguished by the data type of the `fips` value; for the country observations, `fips == US`; for the state observations, `fips` is a character string (the state name); and for the county observations, `fips` is numeric. In general, it's good practice to format data so that each data table contains observations on only one kind of observational unit.

### Question 1 (c)

Separate `election_raw` into separate federal-, state-, and county-level dataframes:

- Store federal-level tallies as `election_federal`.
- Store state-level tallies as `election_state`.
- Store county-level tallies as `election`. Coerce the `fips` variable to numeric.

**(i) Print the first three rows of `election_federal`.** Format the table nicely using `pander()`.

| county | fips | candidate | state | votes |
|---|---|---|---|---|
| NA | US | Donald Trump | US | 62984825 |
| NA | US | Hillary Clinton | US | 65853516 |
| NA | US | Gary Johnson | US | 4489221 |

**(ii) Print the first three rows of `election_state`.** Format the table nicely using `pander()`.

| county | fips | candidate | state | votes |
|---|---|---|---|---|
| NA | CA | Hillary Clinton | CA | 8753788 |
| NA | CA | Donald Trump | CA | 4483810 |
| NA | CA | Gary Johnson | CA | 478500 |

**(iii) Print the first three rows of `election`.** Format the table nicely using `pander()`.

| county | fips | candidate | state | votes |
|---|---|---|---|---|
| Los Angeles County | 6037 | Hillary Clinton | CA | 2464364 |
| Los Angeles County | 6037 | Donald Trump | CA | 769743 |
| Los Angeles County | 6037 | Gary Johnson | CA | 88968 |

**Census data**

The `census` data contains high resolution information (more fine-grained than county-level). In order to align this with the election data, you'll need to aggregate to the county level, which is the highest geographical resolution available in the election data. The following steps will walk you through this process.

**Question 1 (d)**

This first set of initial steps aims to clean up the census data and remove variables that are highly correlated. Write a chain of commands to accomplish the following:

- filter out any rows of `census` with missing values;
- convert `Men`, `Women`, `Employed`, and `Citizen` to percentages of the total population;
- drop `Men`, since the percentage of men is redundant (percent men + percent women = 100)
- compute a `Minority` variable by summing `Hispanic`, `Black`, `Native`, `Asian`, `Pacific` and then remove these variables after creating `Minority`;
- remove `Income`, `Walk`, `PublicWork`, and `Construction`; and
- remove variables whose names end with `Err` (standard errors for estimated quantities).

Store the result as `census_clean`, and print the first 3 rows and 7 columns. Format the printed rows and columns nicely using `pander()`.

| CensusTract | State | County | TotalPop | Women | White | Citizen |
|---|---|---|---|---|---|---|
| 1.001e+09 | Alabama | Autauga | 1948 | 51.75 | 87.4 | 77.16 |
| 1.001e+09 | Alabama | Autauga | 2156 | 50.88 | 40.4 | 77.09 |
| 1.001e+09 | Alabama | Autauga | 2968 | 54.04 | 74.5 | 78.67 |

**Question 1 (e)**

To aggregate the clean census data to the county level, you'll weight the variables by population. Create population weights for sub-county census data by following these steps:

- group `census_clean` by `State` and `County`;
- use `add_tally()` to add a `CountyPop` variable with the population;
- add a population weight variable `pop_wt` computed as `TotalPop/CountyPop` (the proportion of the county population in each census tract);
- multiply all quantitative variables by the population weights (use 'mutate(across(. . . , ~ .x*pop_wt));
- remove the grouping structure (`ungroup()`) and drop the population weights and population variables.

Store the result as `census_clean_weighted`, and print the first 3 rows and 7 columns. Format the output nicely using `pander()`.

| State | County | Women | White | Citizen | IncomePerCap | Poverty |
|---|---|---|---|---|---|---|
| Alabama | Autauga | 1.825 | 3.083 | 2.722 | 907.1 | 0.2857 |
| Alabama | Autauga | 1.987 | 1.577 | 3.01 | 703.6 | 0.9956 |
| Alabama | Autauga | 2.905 | 4.004 | 4.228 | 1112 | 0.6826 |

**Question 1 (f)**

Here you'll aggregate the census data to county level. Follow these steps:

- group the sub-county data `census_clean_weighted` by state and county;
- compute popluation-weighted averages of each variable by taking the sum of each quantitative variable (use `mutate(across(..., sum))`);
- remove the grouping structure.

Store the result as `census_tidy` and print the first 3 rows and 7 columns. Format the output nicely using `pander()`.

| State | County | Women | White | Citizen | IncomePerCap | Poverty |
|---|---|---|---|---|---|---|
| Alabama | Autauga | 51.57 | 75.79 | 73.75 | 24974 | 12.91 |
| Alabama | Baldwin | 51.15 | 83.1 | 75.69 | 27317 | 13.42 |
| Alabama | Barbour | 46.17 | 46.23 | 76.91 | 16824 | 26.51 |

You can check your final result by comparison with the reference dataset in the .Rmd file for this document containing the first 20 rows of the tidy data.

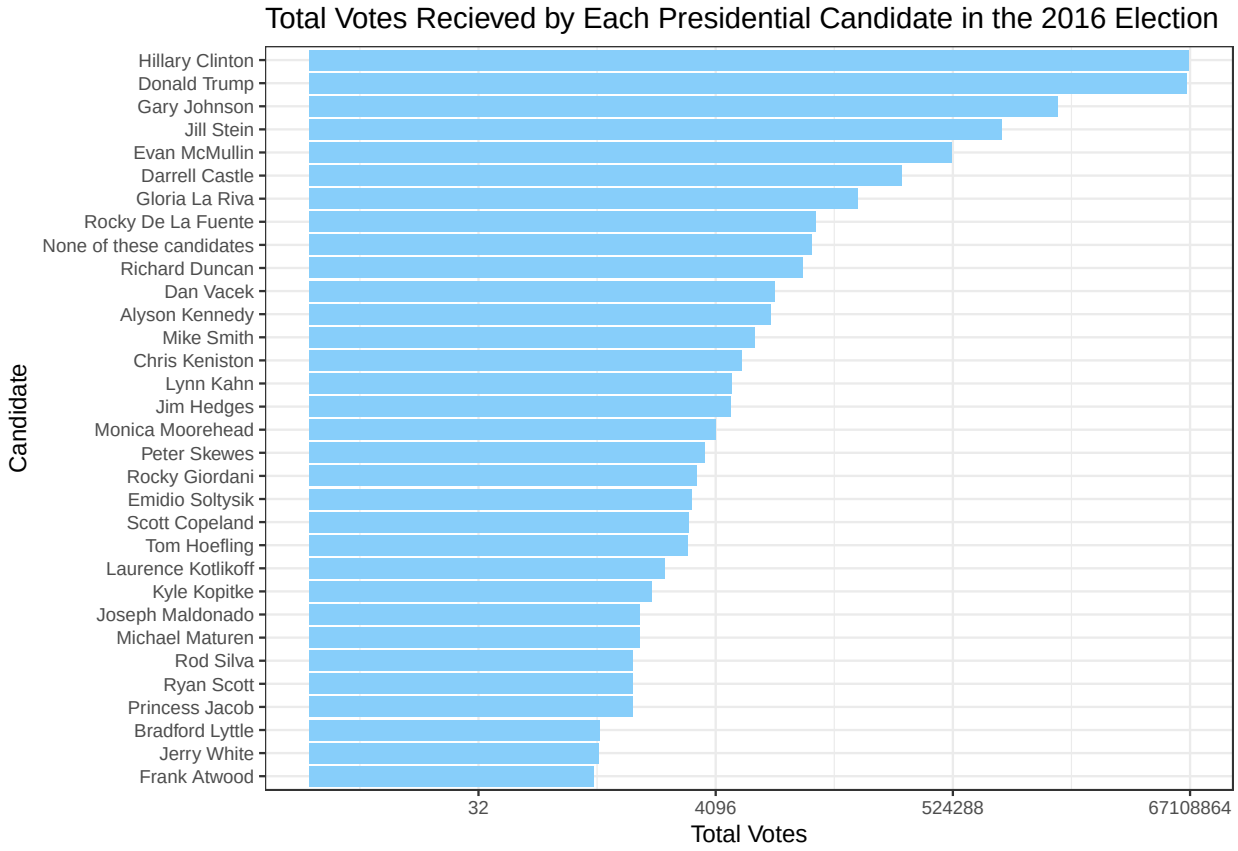
**Question 1 (g)**

Now that you have tidy versions of the census and election data, and a merged dataset, clear the raw and intermediate dataframes from your environment using `rm(list = setdiff(ls(), ...))`. `ls()` shows all objects in your environment, so the command removes the set difference between all objects and ones that you specify in place of `...`; the latter should be a vector of the object names you want to keep. You should keep the three data frames containing election data for the federal, state, and county levels, and the tidy census data.

# Part 2: Exploratory analysis

### Question 2 (a)

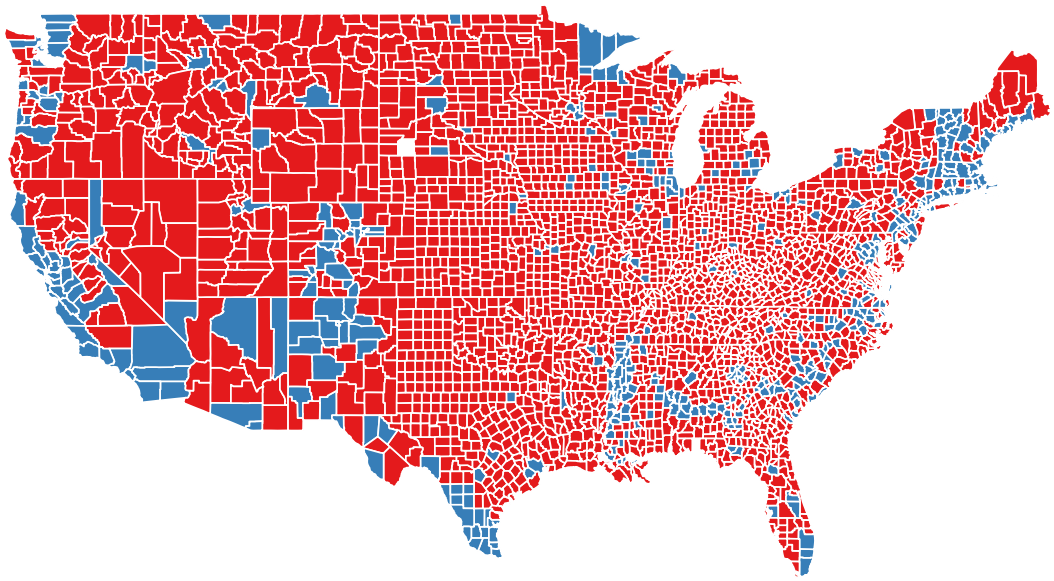
How many named presidential candidates were there in the 2016 election? Draw a bar graph of all votes received by each candidate, and order the candidate names by decreasing vote counts. (*Hints*: use the federal-level election data; you may need to log-transform the vote axis to see all the bar heights clearly.)




Next you'll generate maps of the election data using `ggmap`. The .Rmd file for this document contains codes to generate a map of the election winner by state. The codes retrieve state geographical boundaries and merge the geographic data with the statewide winner found from the election data by state.

### Question 2 (b)

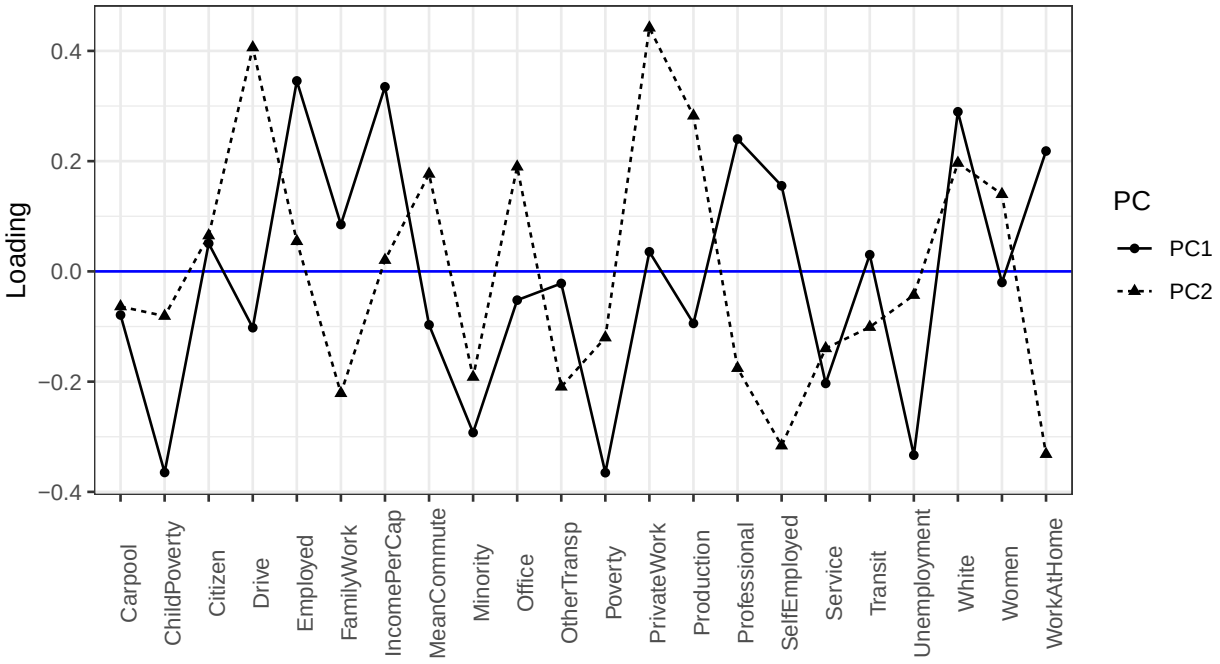
Follow the example above to create a map of the election winner by county. The .Rmd file for this document contains codes to get you started.



**Question 2 (c)**

Which variables drive variation among counties? Carry out PCA for the census data.

**(i) Center and scale the data, compute and plot the principal component loadings for the first two PC's.**

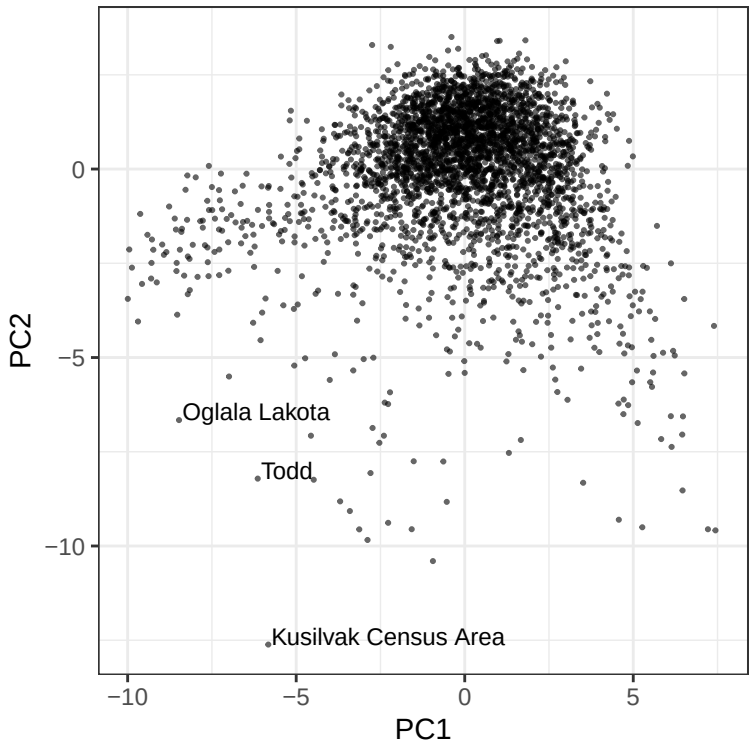



**(ii) Interpret the loading plot. Which variables drive the variation in the data?**
The variables with the highest loading magnitudes drive variation in the data. These variables include working-from-home rate, solo-commuter rate, 'White' and 'Minority' population percentages, poverty rate, self-employment rate, and percent of the workforce in private industry.

General information about income and employment, such as Poverty, Income Per Capita, Employment and Unemployment, corresponds to PC1. Specific professional information, such as Commuter Rate, Work at Home Rate, and Private Work Rate corresponds to PC2.

**(iii) How much total variation is captured by the first two principal components?**
The first two PC's capture 41.44% of the total variation.

**(iv) Plot PC1 against PC2.**

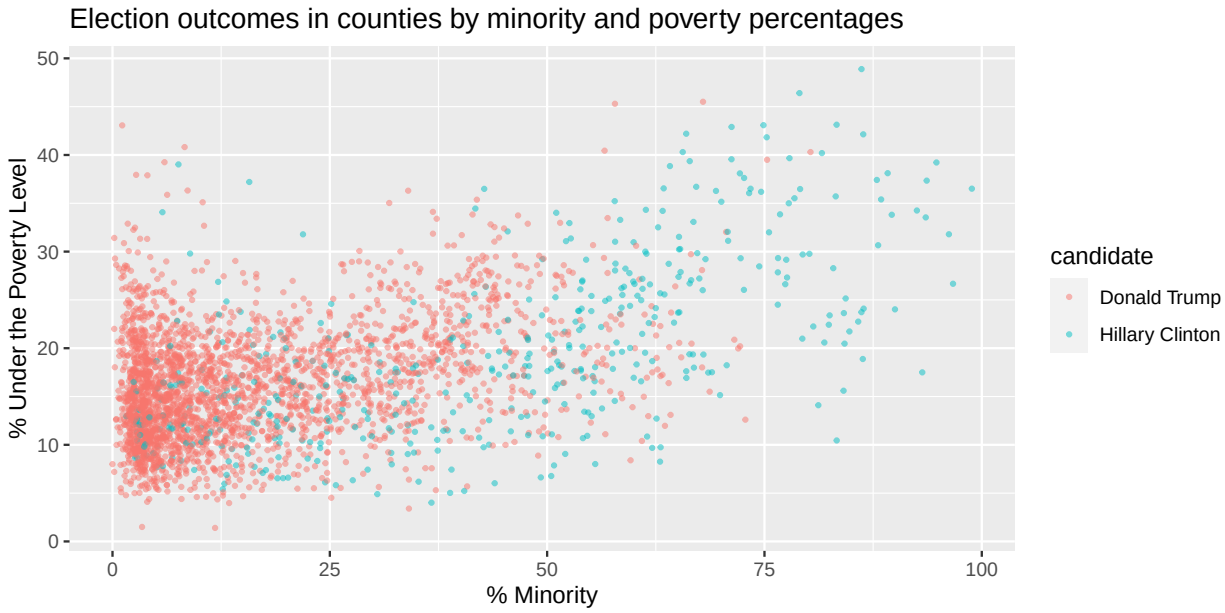



**(v) Do you notice any outlier counties? If so, which counties, and why do you think they are outliers?**

The Kusilvak Census Area is a clear outlier. It has a significantly lower PC2 value than any other county. Oglala Lakota and Todd Counties also have exceptionally low combinations of PC1 and PC2 loading values. Each of these counties have majority Native American populations and exceptionally low income per capita. In fact, they are the three poorest counties in the 50 states according to public data.

**Question 2 (d)**

Create a visualization of your choice using `census` data. Many exit polls noted that demographics played a big role in the election. If you need a starting point, use this Washington Post article and this R graph gallery for ideas and inspiration.


Election outcomes in counties by minority and poverty percentages
% Under the Poverty Level
50
40
30
20
10
0
0
25
50
75
100
% Minority
candidate
Donald Trump
Hillary Clinton

# Part 3: Planned work

Now that you've thought about the prediction problem, tidied and explored the census and election data, you should devise a plan for more focused analysis.

Your objective in the second stage of the project is to analyze a merged county-level dataset. The chunk below this paragraph in the .Rmd file for this document combines the vote information for the winning candidate and runner-up in each county with the census data.

There are a number of possibilities for analyzing this data. Here are just a few:

- Prediction
  - Predict the winner of the popular vote
  - Predict the winner of the general election
  - Predict the winner of each county
  - Predict the vote margin by county
  - Predict the vote margin by state
- Inference
  - Model the probability one candidate wins a county and identify significant associations with census variables
  - Model the vote margin and identify/interpret significant associations with census variables
  - Cluster or group counties and model the probability of a win by one candidate or the vote margin separately for each cluster; look for different patterns of association
  - Model the relationship between votes (or win probabilities) separately for each candidate, and contrast the results.

Each would require some slightly different preprocessing of `merged_data` to select the relevant rows and columns for the specified tasks.

**Question 3**

Propose an analysis that you'd like to carry out. Be specific: indicate two tasks you'll pursue and for each task indicate the methods you'll use to approach the task. Your methods description should include mention of how you will prepare `merged_data` for modeling, and which model(s) you'll try.

These descriptions don't need to be long, just enough to convey the general idea. Also, these are not final commitments – you can always change your mind later on if you like.

**Task 1** **Task**: Based on the given data we try to predict the winner of general election 2016 in each state and check how the model performs.

**Methods**: Since we want to classify the winning states for the winning candidate, we are looking for the best classification method among the following: 1) Logistic regression. 2) K nearest neighbor (KNN). 3) Linear Discriminant Analysis. 4) Quadratic discriminant analysis. 5) Regression Tree.

Then we check which method performs best.

**Task 2** **Task**: Since the outcome of election is predicted based on covariates, we may try an optimal clustering of those points in feature space. That way we may try to explain the varibility in election performance.

**Methods**: Clustering method.