1. BRIEF DESCRIPTION OF THE DATA SETS AND A SUMMARY OF THEIR ATTRIBUTES

First dataset: election results by county from U.S. Senate elections in 2022. Source: https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/YB60EJ

Contains all vote totals by county for all elections to US senate in 2022. Attributes include name of state, name of county, county id, name of candidate, detailed party description, simplified party description, number of votes, mode of voting (election day, mail, etc.)

Second dataset: economic parameters of U.S counties. Source: https://www.ers.usda.gov/data-products/county-level-data-sets/county-level-data-sets-download-data/

Contains different values of economic parameters for all U.S. states and counties, including unemployment rate in 2022, median household income in 2021, and percent of median household income divided by the statewide median household income in 2021.

2. INITIAL PLAN FOR DATA EXPLORATION

Sum election data for every county into three categories: votes for the mainstream Democratic party candidate, votes for the mainstream Republican party candidate, and other votes. Merge datasets and explore correlations between economic and election data.

3. ACTIONS TAKEN FOR DATA CLEANING AND FEATURE ENGINEERING

Election dataset cleaning :

First, I described the dataset using info function. Then I checked for columns which didn’t seem useful using pandas unique() function. I deleted the data for a runoff election in Georgia and a special election in Oklahoma, so that every state in the dataset would have only one election. Then, I dropped columns which were not useful.

```
[3]: senate.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21618 entries, 0 to 21617
Data columns (total 21 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   year              21618 non-null  int64
 1   date              21618 non-null  object
 2   state             21618 non-null  object
 3   state_po          21618 non-null  object
 4   state_fips        21618 non-null  int64
 5   state_cen         21618 non-null  int64
 6   state_ic          21618 non-null  int64
 7   county_name       21316 non-null  object
 8   county_fips       21316 non-null  float64
 9   office            21618 non-null  object
 10  candidate         21300 non-null  object
 11  party_detailed    19376 non-null  object
 12  party_simplified  21147 non-null  object
 13  writein           21618 non-null  bool
 14  candidatevotes    21618 non-null  float64
 15  totalvotes        21618 non-null  int64
 16  unoffical         21618 non-null  bool
 17  stage             21618 non-null  object
 18  special           21618 non-null  bool
 19  mode              21618 non-null  object
 20  version           21618 non-null  int64
dtypes: bool(3), float64(2), int64(6), object(10)
memory usage: 3.0+ MB
```

```
senate.info()

<class 'pandas.core.frame.DataFrame'>
Index: 19700 entries, 0 to 21617
Data columns (total 12 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   state             19700 non-null  object
 1   state_po          19700 non-null  object
 2   state_fips        19700 non-null  int64
 3   county_name       19398 non-null  object
 4   county_fips       19398 non-null  float64
 5   candidate         19541 non-null  object
 6   party_detailed    17617 non-null  object
 7   party_simplified  19229 non-null  object
 8   writein           19700 non-null  bool
 9   candidatevotes    19700 non-null  float64
 10  totalvotes        19700 non-null  int64
 11  mode              19700 non-null  object
dtypes: bool(1), float64(2), int64(2), object(7)
memory usage: 1.8+ MB
```

Then I found states that don’t issue data by county using senate[senate.county_name.isnull()].state.value_counts(). There were only statewide vote totals for Pennsylvania, Alaska and Vermont, and since I planned to analyse data by county, I deleted data for those states.

Then I looked at states that provide data by different modes of voting using senate[senate['mode']!='TOTAL'].state.value_counts(). Those states were North Carolina, Georgia, Iowa, Arkansas and South Carolina. Those states had a separate row for each candidate, county and mode of voting. Looking at them separately, I found that Georgia, Iowa, Arkansas and South Carolina had rows with all votes for each candidate and county, but North Carolina only had data by each mode of voting.

I iterated over each candidate+county in North Carolina, summed their votes for all modes of voting, and then added necessary rows to the dataset (all code is in the files). Then I deleted all rows detailing votes by mode of voting and dropped the ‘mode’ column.

Then, I dealt with party names first by checking senate.party_detailed.value_counts(). I wanted to drop detailed descriptions of parties. States of Illinois and Maryland had only candidates from ‘Democratic’ party, but in the party_simplified column they were marked as ‘Other’. I marked the only Democraic candidate from Illinois as democrat, and marked one of two democratic candidates with by far the most votes as democrat in Maryland. Also, in New York major candidates ran from two parties each.

```
[67]: senate[senate.state=='NEW YORK'].groupby(['candidate','party_detailed']).candidatevotes.sum()

[67]: candidate           party_detailed
      CHARLES E. SCHUMER  DEMOCRAT            3022822.0
                          WORKING FAMILIES     297739.0
      DIANE SARE          LAROUCHE              26844.0
      JOE PINION          CONSERVATIVE         296652.0
                          REPUBLICAN          2204499.0
      Name: candidatevotes, dtype: float64
```

I found that using a filter, grouping data by candidate and party and then summing votes. I iterated over rows and for each county, added WORKING FAMILIES party votes for CHARLES E. SCHUMER to his democrat total, and added CONSERVATIVE votes for JOE PINION to his republican rows. Then I deleted rows for these CONSERVATIVE and WORKING FAMILIES parties in New York.

Then I looked for counties where there were not candidates for each major parties. I found all of them consisted the states of Utah and Missouri. For some reason, in Missouri parties were not marked, so I added the party for democrat and republican candidates after searching for that election online. In Utah, Democratic party did not run a candidate in their U.S. Senate election in 2022, but endorsed an independent candidate instead. Therefore, I had to keep in mind that Utah data could only be used for analyzing republican votes.

Then I checked for remaining NULL values in candidate names. All of those values were in Georgia, which was just a weird way of keeping total votes cast in each county. I deleted those rows.

Then I checked for remaining NULL values in the ‘parties’ column. That is how I found out that various states, for some reason, included data about overvotes and undervotes, which are not valid votes. I deleted those rows. Also, Nevada included data about votes for ‘none of these candidates’, which I counted as third party votes.

After that, I filled remaining NULL party values with ‘OTHER’ and finally dropped the party_detailed column.

Now I needed to have just one entry for democrats and one for republicans in every county (apart from the 29 counties of Utah). Using senate.party.value_counts(), I found there were still extra entries.

I changed libertarian party identifications to ‘other’ and then grouped by state, candidate name and total counties only for dem or rep candidates, hoping to find two candidates with equal number of counties for each state (apart from Utah). That was not the case for Arizona and Louisiana.

```
senate[senate.state=='ARIZONA'].groupby(['party','candidate']).candidatevotes.sum()

party  candidate
DEM    MARK KELLY                            1322027.0
       TODD JAMES SMELTZER                         6.0
       TY RICHARD MCLEAN JR.                      21.0
       WILLIAM "WILL" MICHAEL TAYLOR               8.0
OTH    LESTER "SKIP" MAUL                         95.0
       MARC J. VICTOR                          53762.0
REP    BLAKE MASTERS                         1196308.0
       CHRISTOPHER BULLOCK                        27.0
       EDWARD DAVIDA                               3.0
       ROXANNE RENEE RODRIGUEZ                    20.0
       SHERRISE BORDES                            17.0
Name: candidatevotes, dtype: float64
```

I found that in Arizona, there were many minor candidates for both Republican and Democratic parties, probably write-ins. I reclassified them as third-party, leaving only Mark Kelly for Democrats and Blake Masters for Republicans.

```
[125]: senate[senate.state=='LOUISIANA'].groupby(['party','candidate','writein']).candidatevotes.sum()

[125]: party  candidate                  writein
       DEM    "LUKE" MIXON               False      182887.0
              GARY CHAMBERS, JR.         False      246933.0
              MV "VINNY" MENDOZA         False       11910.0
              SALVADOR P. RODRIGUEZ      False        7767.0
              SYRITA STEIB               False       31568.0
       OTH    "XAN" JOHN                 False        2753.0
              AARON C. SIGLER            False        4865.0
              BERYL A. BILLIOT           False        9378.0
              BRADLEY MCMORRIS           False        5388.0
              THOMAS WENN                False        1322.0
              W. THOMAS LA FONTAINE OLSON False       1676.0
       REP    DEVIN LANCE GRAHAM         False       25275.0
              JOHN KENNEDY               False      851568.0
       Name: candidatevotes, dtype: float64
```

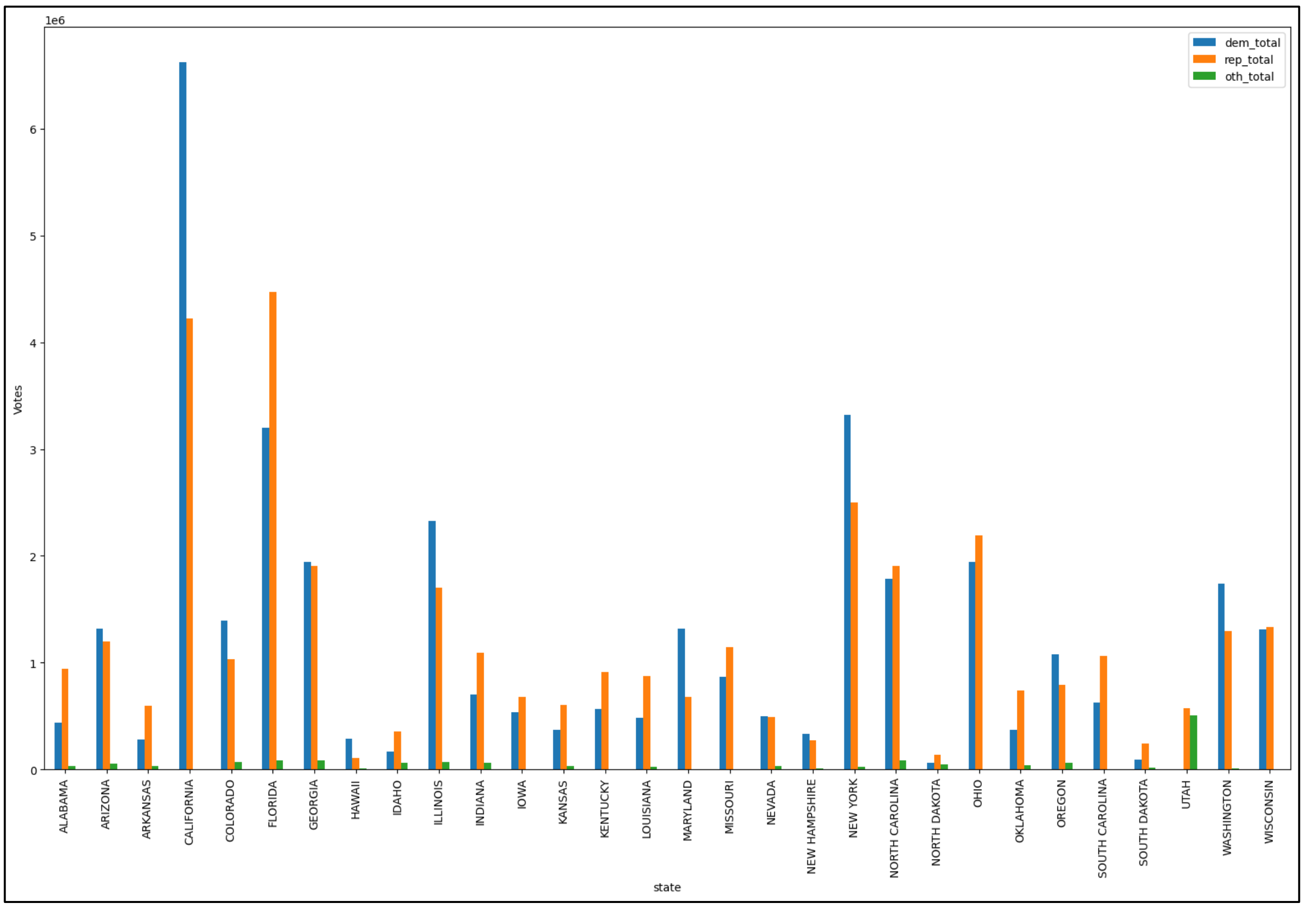
However, in Louisiana there were no dominating candidates, especially for Democrats, since Louisiana does not hold primaries for U.S. Senate elections. So I iterated over all counties in Louisiana and summed the vote totals for republicans, democrats and others, deleted all Louisiana rows from dataset and added new summed ones.

And even after all of that, I still did not get the dataset in the right condition. I looked at duplicates and found many, all from Indiana. For some reason, votes from Indiana were provided not by county, but by precinct – the smallest electoral division. I summed all the votes, deleted old rows and added new ones for Indiana.

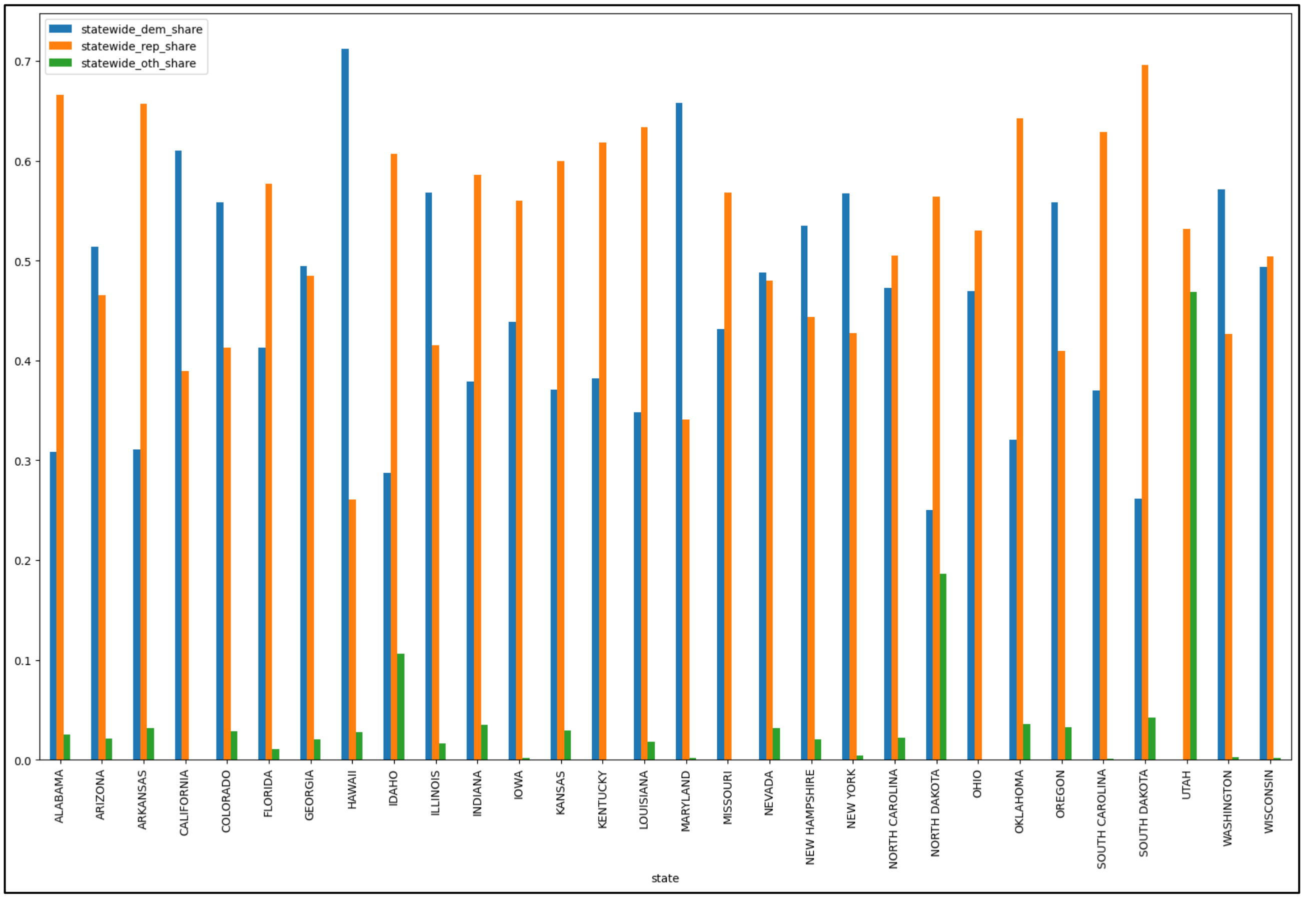
After that, I was finally good to go, and I summed all the third-party votes for each county in the dataset. I got a dataset with each county having either 2 or 3 entries, depending on whether there were any third-party votes.

Election dataset feature engineering:

First, I wanted to look at the votes statewide. I summed votes for each party and stat, created a separate dataframe with total votes cast in the state. Then I visualized it using Pandas' version of Matplotlib.



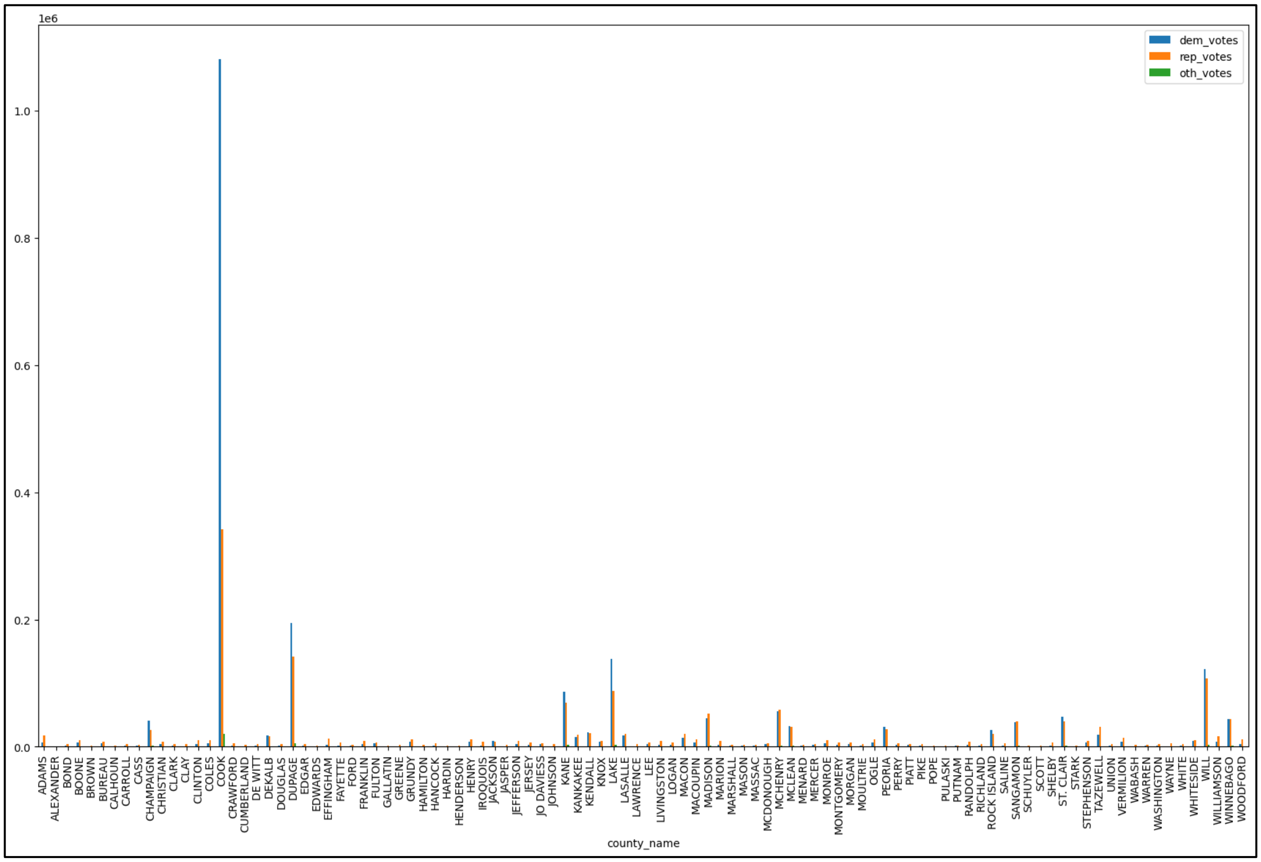
As is seen on the map, U.S. states have vastly different populations and vote totals. So vote totals themselves don't tell much, ehat matters is the rate of votes. I added statewide dem and rep ratios of vote, as well as a dem/rep ratio, a third-party ratio, and a Boolean variable telling who won the race in the state. Visualizing some of the rates:



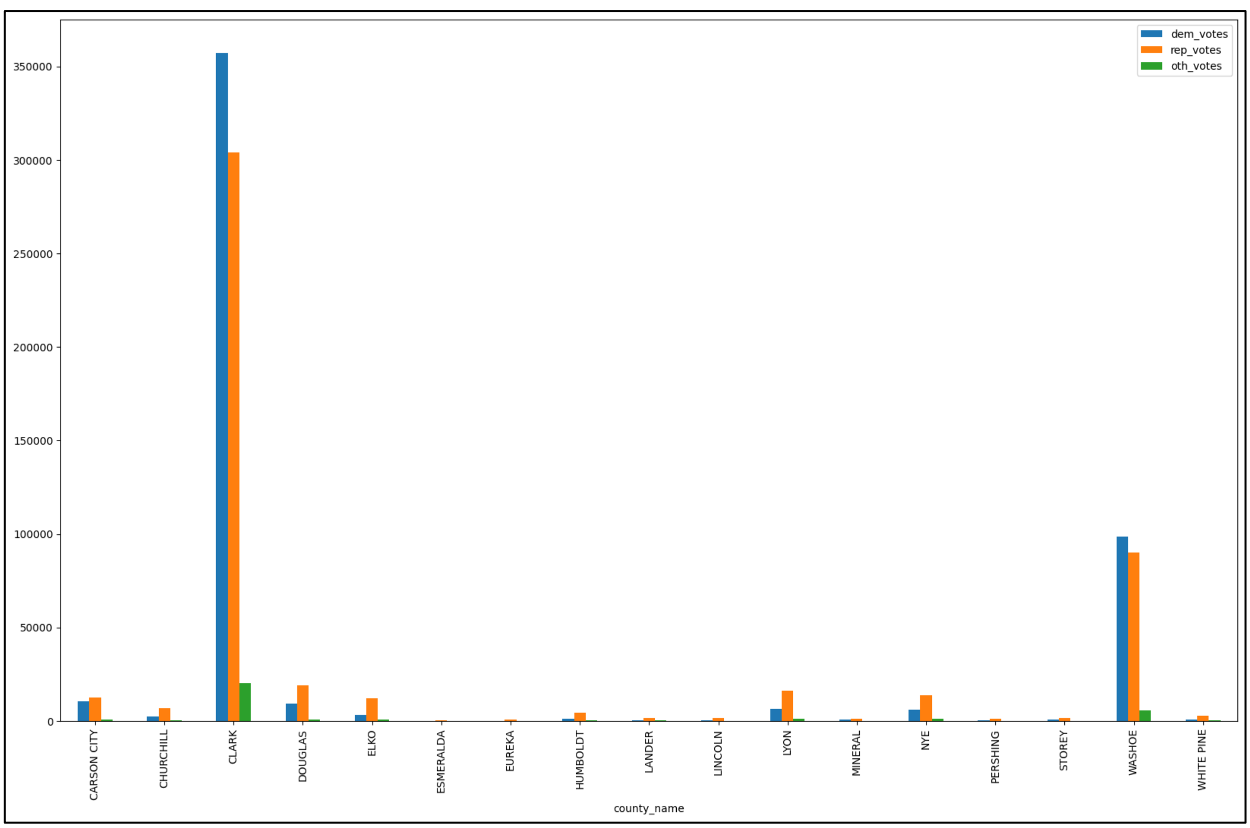
Obviously, vote rates are much more uniform and therefore useful for modeling.

Then I created a new ‘county-wide’ dataset with one entry for each county and dem, rep and oth votes in separate columns.

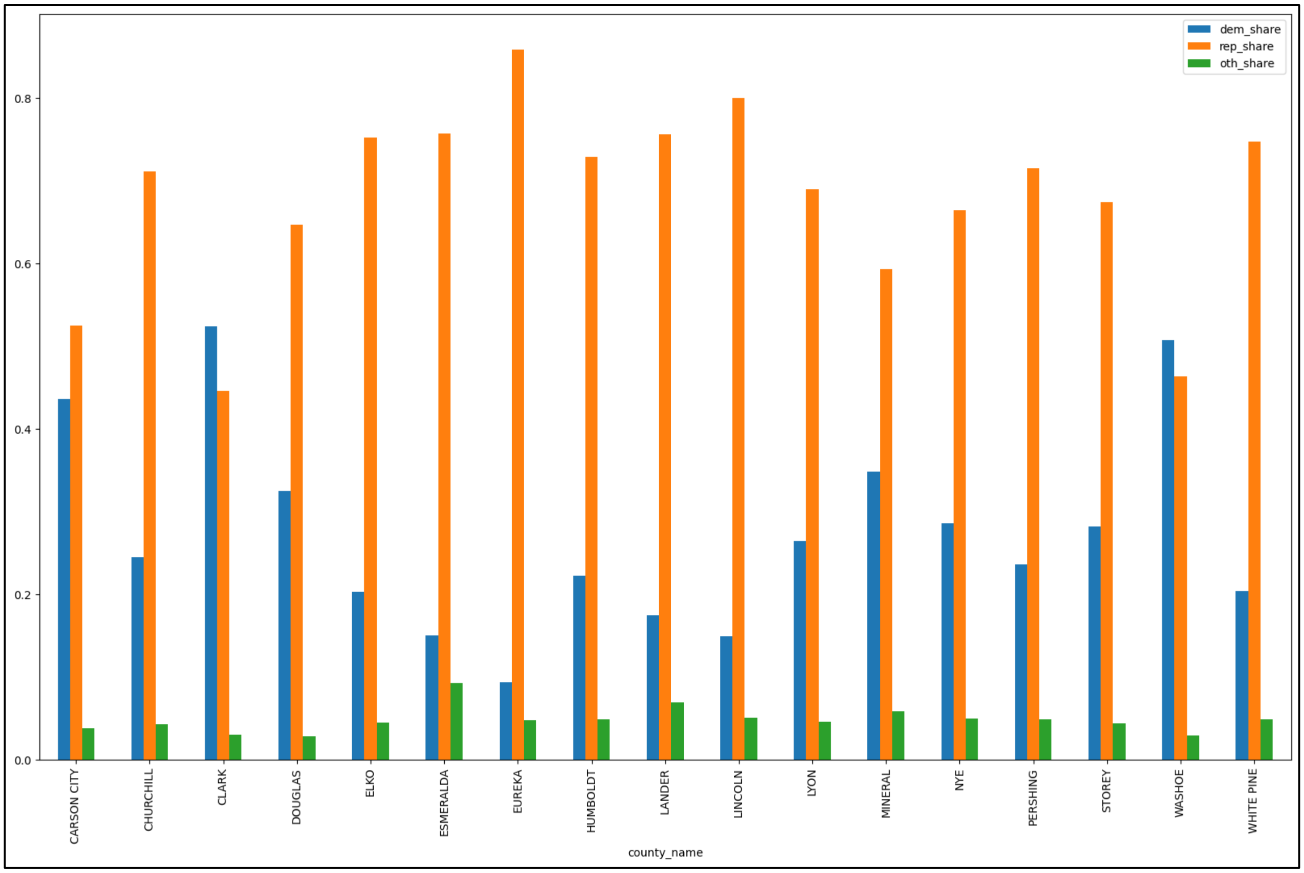
In some states, there is one county that houses most and the voters, for example in Illinois:


1e6
1.0
0.8
0.6
0.4
0.2
0.0
dem_votes
rep_votes
oth_votes
ADAMS
ALEXANDER
BOND
BOONE
BROWN
BUREAU
CALHOUN
CARROLL
CASS
CHAMPAIGN
CHRISTIAN
CLARK
CLAY
CLINTON
COLES
COOK
CRAWFORD
CUMBERLAND
DE WITT
DEKALB
DOUGLAS
DUPAGE
EDGAR
EDWARDS
EFFINGHAM
FAYETTE
FORD
FRANKLIN
FULTON
GALLATIN
GREENE
GRUNDY
HAMILTON
HANCOCK
HARDIN
HENDERSON
HENRY
IROQUOIS
JACKSON
JASPER
JEFFERSON
JERSEY
JO DAVIESS
JOHNSON
KANE
KANKAKEE
KENDALL
KNOX
LAKE
LASALLE
LAWRENCE
LEE
LIVINGSTON
LOGAN
MACON
MACOUPIN
MADISON
MARION
MARSHALL
MASON
MASSAC
MCDONOUGH
MCHENRY
MCLEAN
MENARD
MERCER
MONROE
MONTGOMERY
MORGAN
MOULTRIE
OGLE
PEORIA
PERRY
PIATT
PIKE
POPE
PULASKI
PUTNAM
RANDOLPH
RICHLAND
ROCK ISLAND
SALINE
SANGAMON
SCHUYLER
SCOTT
SHELBY
ST. CLAIR
STARK
STEPHENSON
TAZEWELL
UNION
VERMILION
WABASH
WARREN
WASHINGTON
WAYNE
WHITE
WHITESIDE
WILL
WILLIAMSON
WINNEBAGO
WOODFORD
county_name

Or Nevada:


350000
300000
250000
200000
150000
100000
50000
0
dem_votes
rep_votes
oth_votes
CARSON CITY
CHURCHILL
CLARK
DOUGLAS
ELKO
ESMERALDA
EUREKA
HUMBOLDT
LANDER
LINCOLN
LYON
MINERAL
NYE
PERSHING
STOREY
WASHOE
WHITE PINE
county_name

Therefore, I had to create rates similar to statewide ones for each county, including a bool value for who won each county. That is what they looked like afterwards in Nevada.



Then I merged the original dataset with the statewide one, adding statewide data to each column. Afterwards, I added ratios of the described features of county value divided by value of that feature statewide. Compensating for the fact that states have different number of counties, I added the feature of county weight=((total votes in county)/(total votes in state))*(number of counties in state). I added same features for dem and rep votes separately. In conclusion, I had following features:

```
[69]: senate.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1968 entries, 0 to 1967
Data columns (total 30 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   state                           1968 non-null   object
 1   state_po                        1968 non-null   object
 2   county_name                     1968 non-null   object
 3   county_fips                     1968 non-null   float64
 4   total_votes                     1968 non-null   float64
 5   dem_votes                       1968 non-null   float64
 6   rep_votes                       1968 non-null   float64
 7   oth_votes                       1968 non-null   float64
 8   dem_name                        1968 non-null   object
 9   rep_name                        1968 non-null   object
 10  dem/rep                         1968 non-null   float64
 11  dem_share                       1968 non-null   float64
 12  rep_share                       1968 non-null   float64
 13  oth_share                       1968 non-null   float64
 14  county_winner                   1968 non-null   object
 15  statewide_total_votes           1968 non-null   float64
 16  statewide_dem_votes             1968 non-null   float64
 17  statewide_rep_votes             1968 non-null   float64
 18  statewide_oth_votes             1968 non-null   float64
 19  winner                          1968 non-null   object
 20  statewide_dem/rep               1968 non-null   float64
 21  statewide_dem_share             1968 non-null   float64
 22  statewide_rep_share             1968 non-null   float64
 23  statewide_oth_share             1968 non-null   float64
 24  dem/rep_ratio_to_statewide      1968 non-null   float64
 25  total_votes_ratio_to_statewide  1968 non-null   float64
 26  counties_per_state              1968 non-null   int64
 27  county_weight                   1968 non-null   float64
 28  dem_votes_ratio_to_statewide    1939 non-null   float64
 29  rep_votes_ratio_to_statewide    1968 non-null   float64
dtypes: float64(22), int64(1), object(7)
```

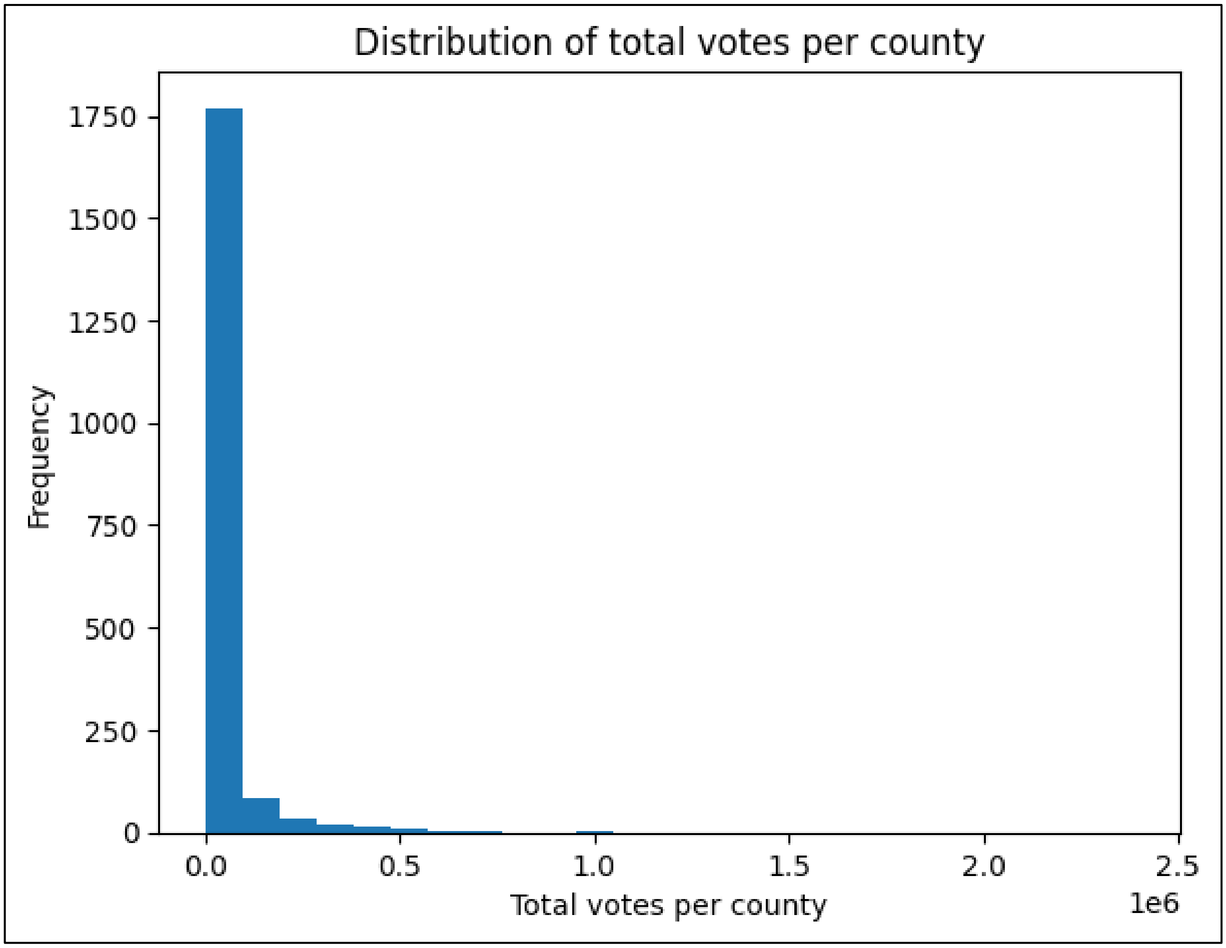
Economics dataset:

Originally, this dataset had only columns identifying county, name of economic parameter, and its value. Through filtering, creating separate dataframes and merging, I got the dataframe with one row per county, and columns of median household income, median household income ratio to statewide value, and unemployment rate. I added the column for unemployment rate ratio to statewide value, then merged two datasets together and dropped NULLs, with only counties with both election and economic data remaining.

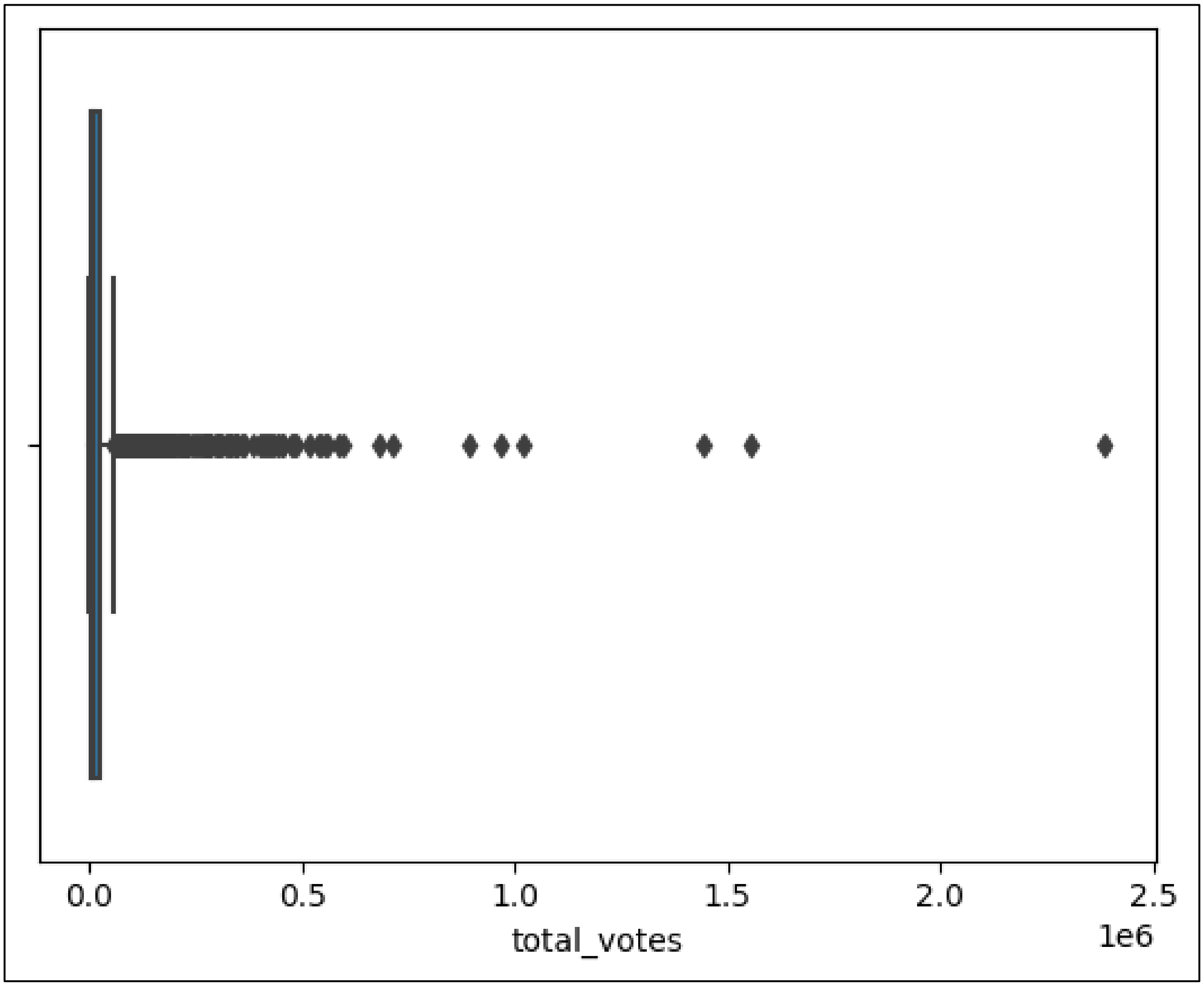
4. KEY FINDINGS AND INSIGHTS, WHICH SYNTHESIZES THE RESULTS OF EXPLORATORY DATA ANALYSIS IN AN INSIGHTFUL AND ACTIONABLE MANNER

I explored relationship between county total votes and party winning that county. I clearly found that Democrats succeeded more in large counties, and republicans – in small ones. However, there were very few large counties.

Here's a histogram of total votes per county. By far most counties have fewer than 100.000 votes.

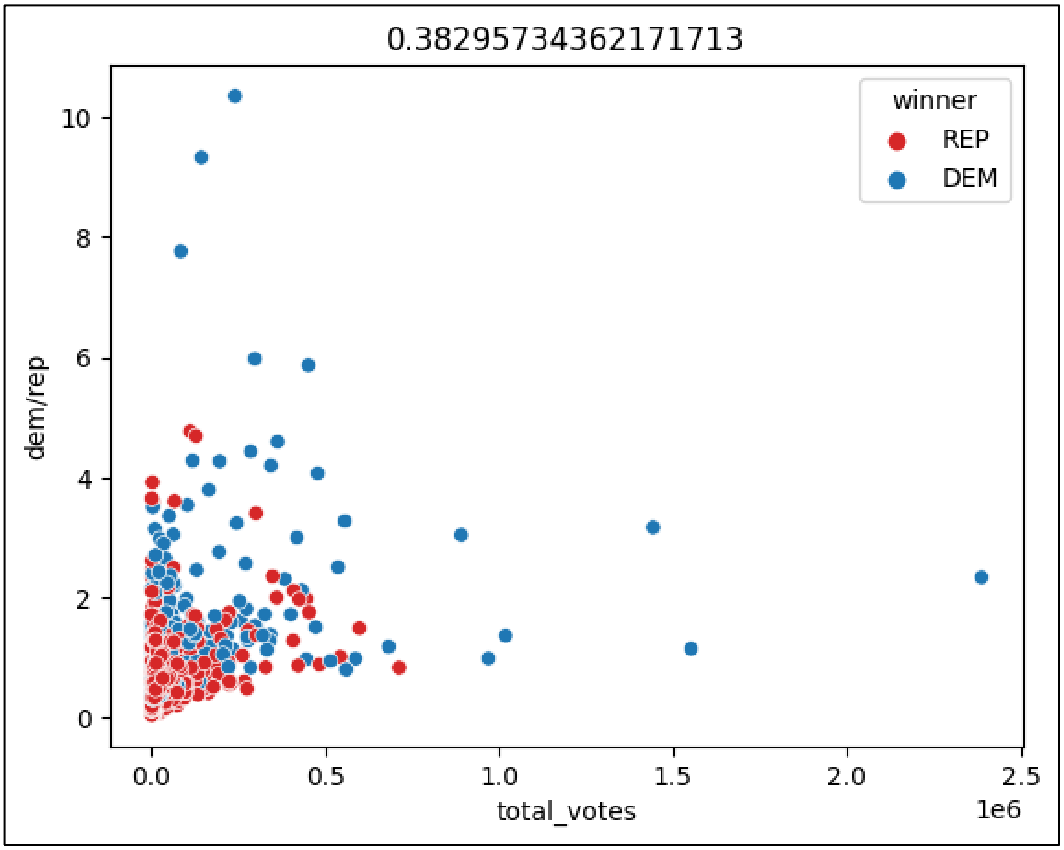


Here is the same parameter in the seaborn box plot.

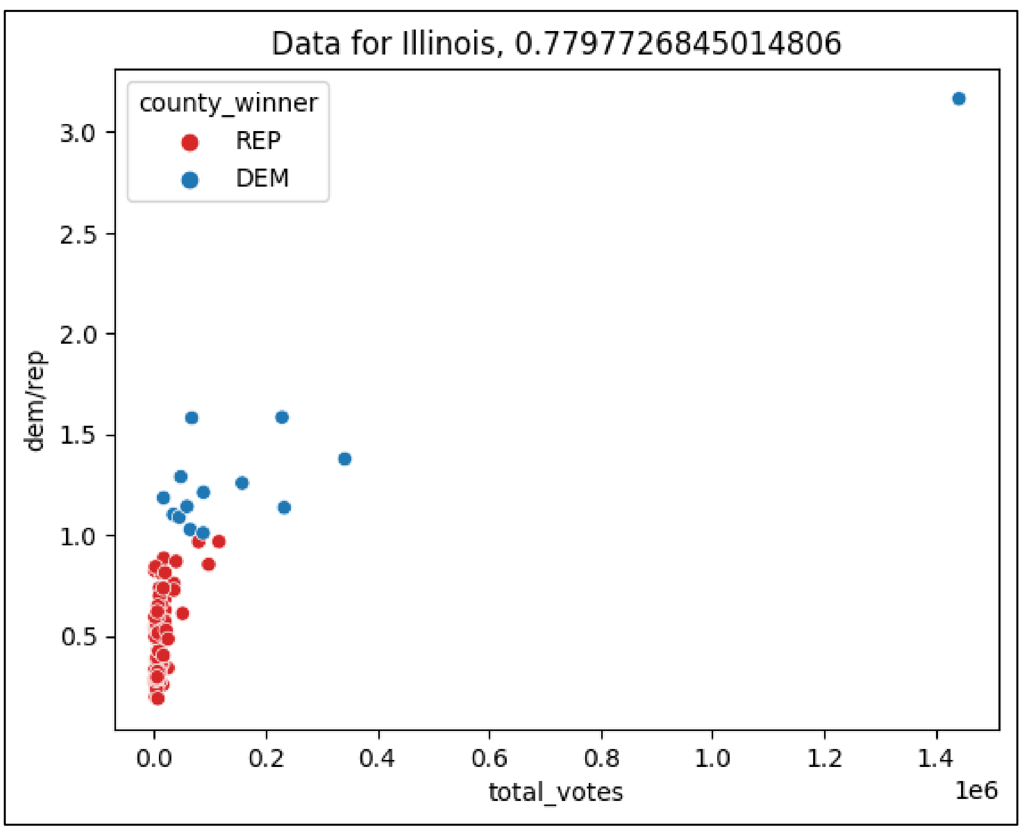


Those counties are definitely outliers, but, of course, they cannot be deleted, because as was seen in visualizations previously, many if not most of Democratic votes in Illinois came from the large Cook County, or in Nevada from Clark County.

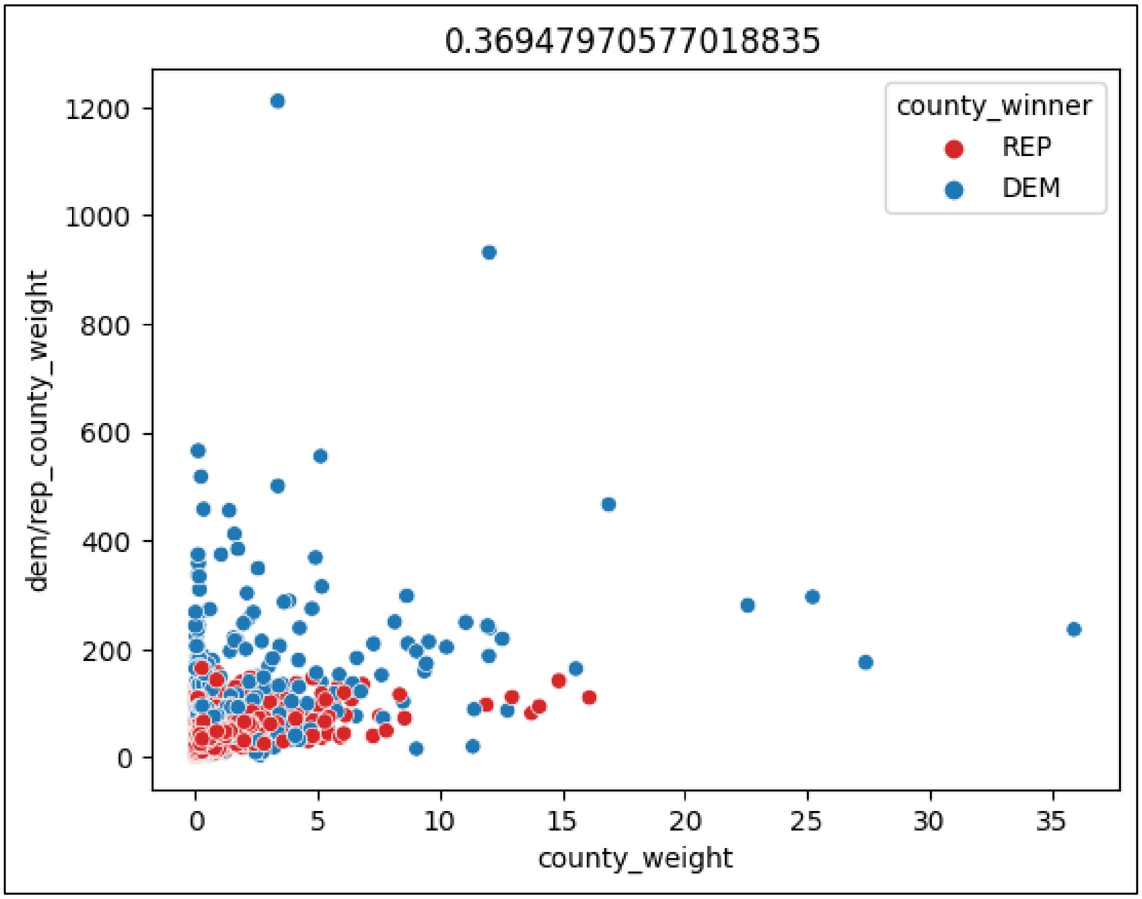
And here is a seaborn scatter plot showing counties' total votes and share of dem/rep vote. Counties from states won by Democrats are colored in blue, from states won by republicans – in red. In the title, there's a correlation coefficient of 0.38 between these values. It's not a strong correlation, but it definitely exists, and we can see that all the counties with more than 600,000 or so votes are from 'blue' states, and all of them individually were won by Democrats, since the rate for them is higher than 1. So it's safe to say that those counties played a big role in multiple election results.



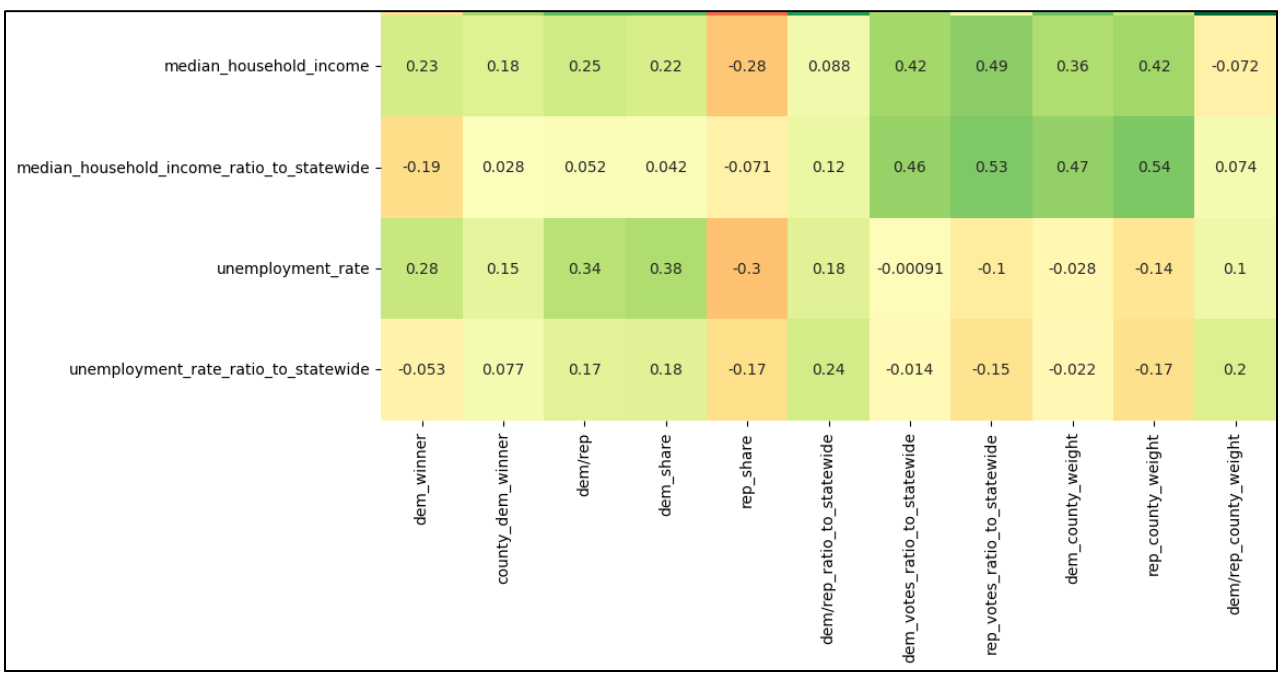
If we take data just from Illinois, the positive correlation between Democratic success and total votes per county is strong.



Correlation between some other parameters also points to the same conclusion.

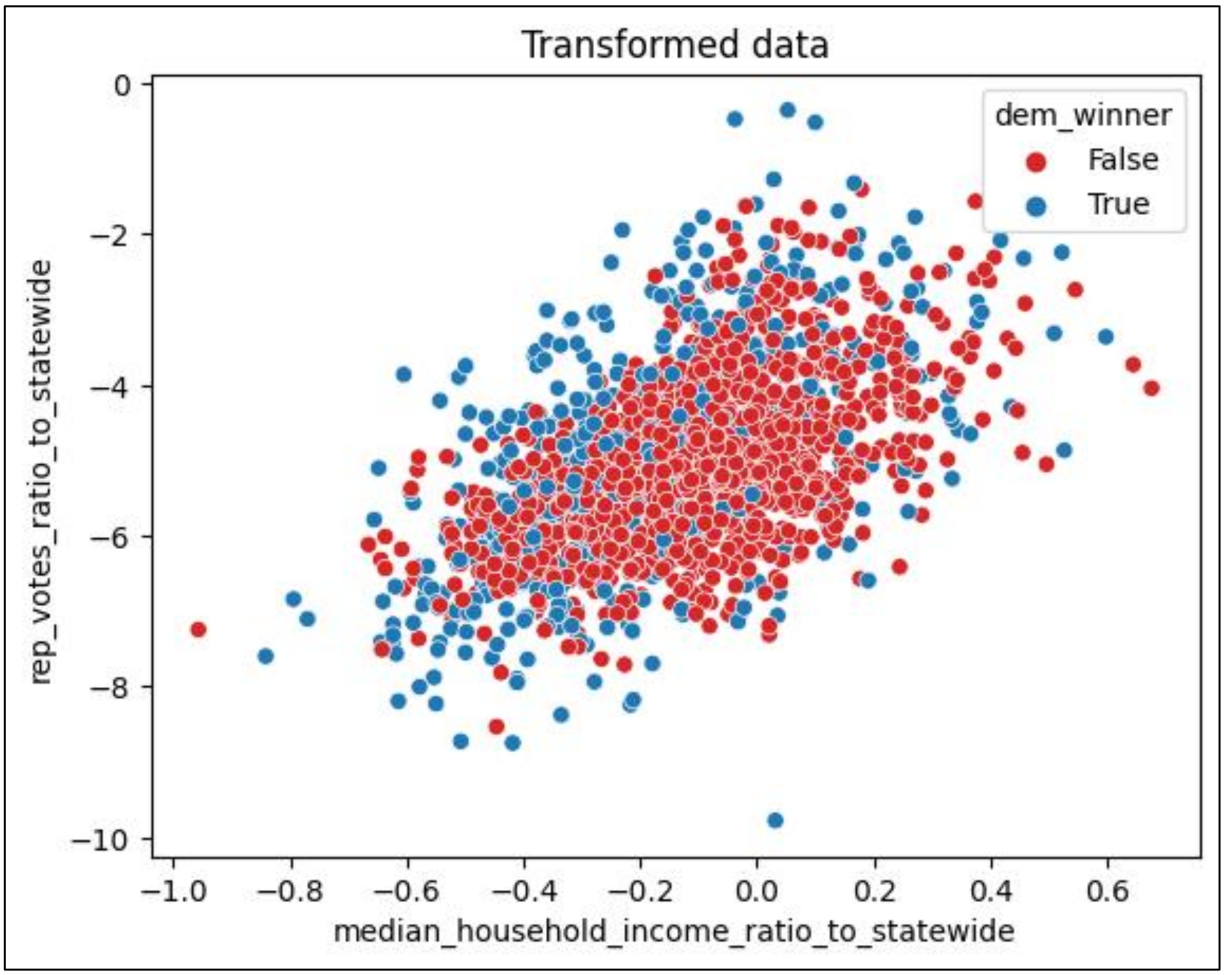


Here is the analysis of correlations between log-transformed economic and election parameters using a seaborn heatmap.



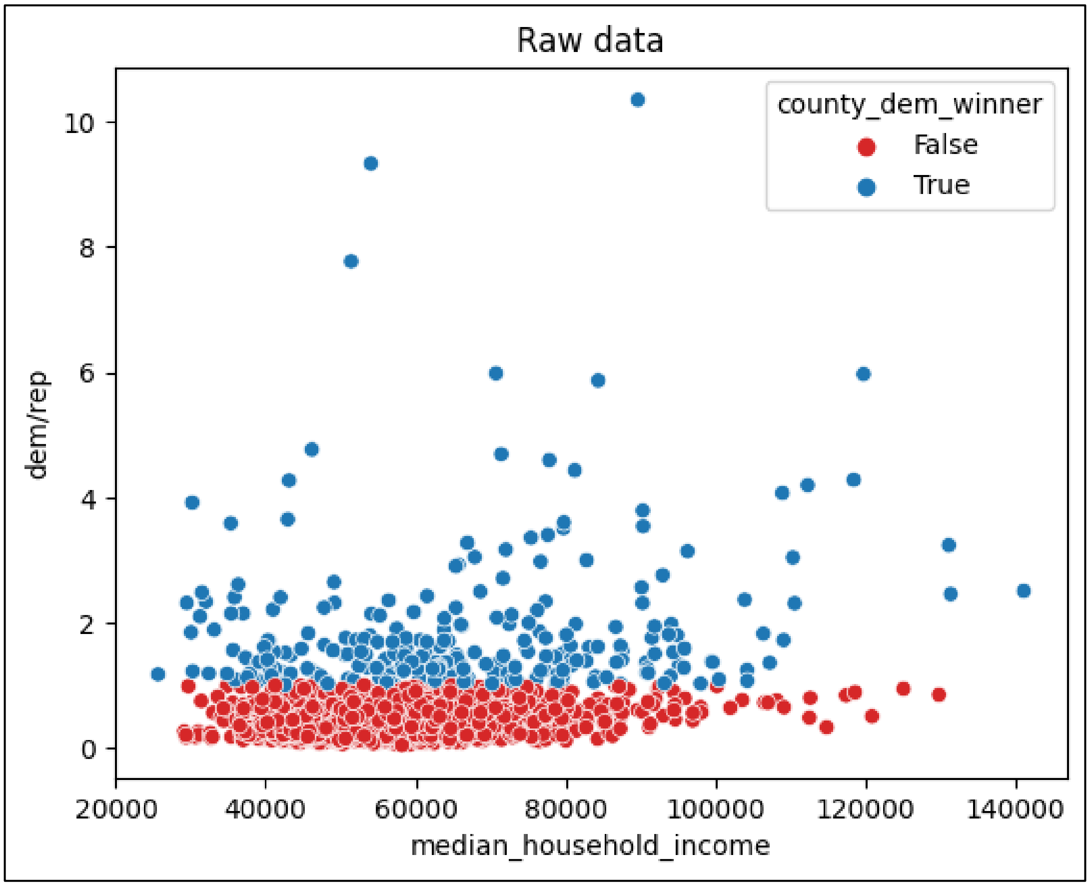
As we can see, there are no strong correlations here. However, there are a few moderately strong ones. For example, the top right corner points to positive correlations between median household income (both absolute and relative to statewide) and parameters describing vote shares for both democrats and republicans relative to their statewide ratio. From this, it could be concluded that within any state on average, people in wealthier counties tend to vote more for one of the main parties, while in poorer counties relative to the state, third-party vote share is higher, relative to the state.

Here is the scatterplot of one of these relationships.



Looking to the left side of the correlation table, there is a weak-to-moderate positive correlation between various metrics of democratic success and both median household income and unemployment rate. This is peculiar, since median household income and unemployment rate themselves unsurprisingly have a definite negative relationship with a correlation of -0.29, climbing to -0.48 if both parameters are taken as a ratio to statewide value. Therefore, if the county is richer or with higher unemployment, it is more likely to vote democratic, but if the county is richer, the unemployment is likely to be lower and vice versa.

Here is one of these relationships illustrated using raw data.



The correlation is not strong, having a coefficient of 0.27, but it exists.

Generally, the correlation structure in raw data is similar to transformed data, only the correlations themselves are smaller, but not by a large margin.

5. FORMULATING AT LEAST 3 HYPOTHESIS ABOUT THIS DATA

After conducting data analysis, I formulated the following 3 hypotheses.

1. In counties won by democrats, median household income is higher.
2. There is difference between values of total votes in counties won by democrats and republicans.
3. In counties with ratio of median household income to statewide value higher than 1.0, unemployment rates are lower than in the other counties.

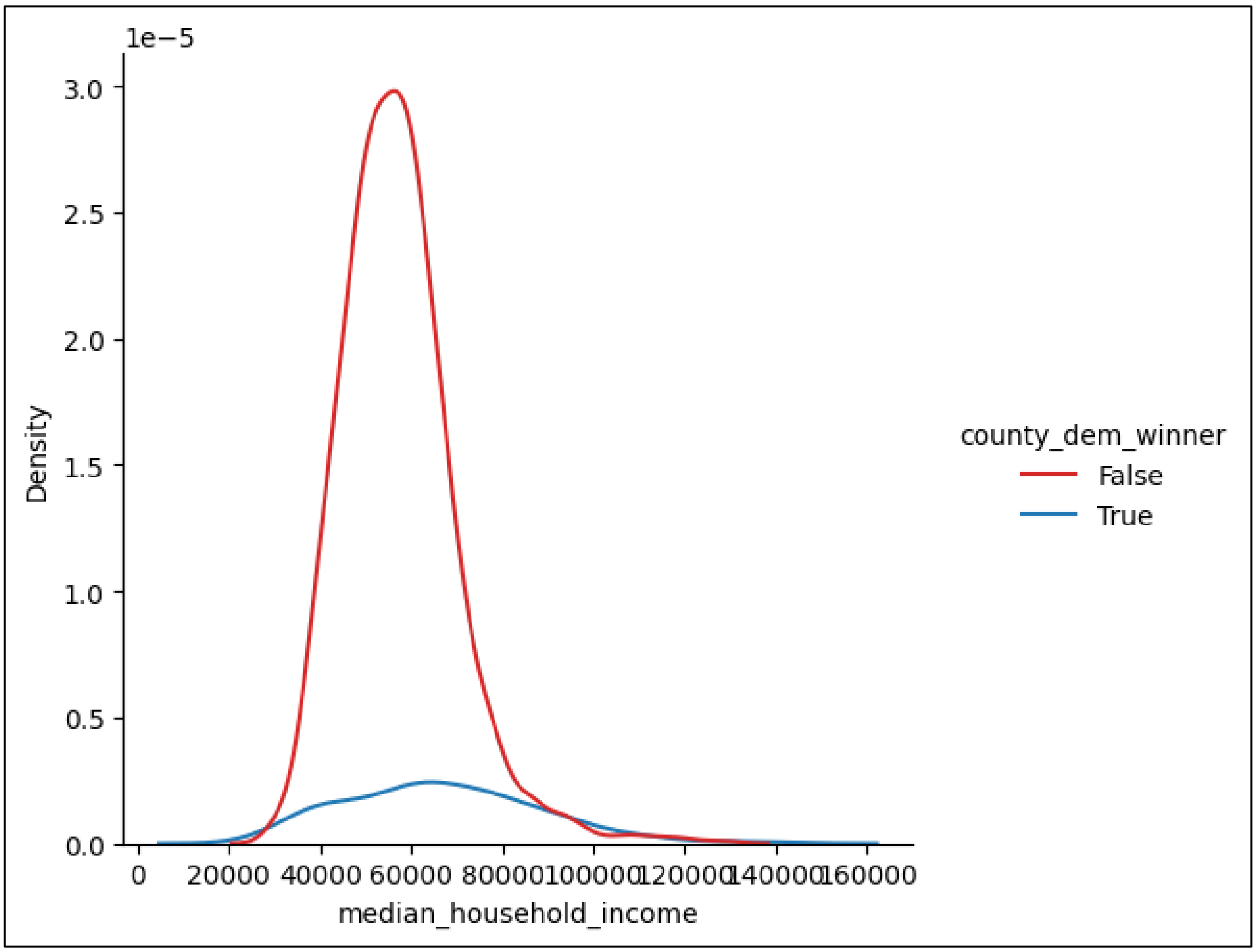
6. CONDUCTING A FORMAL SIGNIFICANCE TEST FOR ONE OF THE HYPOTHESES AND DISCUSS THE RESULTS

Let us test the first hypothesis. First, we formulate its null and alternative.

H0: Median household income values in counties won by democrats are less or equal than in counties won by republicans.

H1: Median household income values in counties won by democrats are higher than in counties won by republicans.

Let us look at a seaborn plot of a smooth distribution function of medium household income values by winner of the vote in that county.



We can see that both samples are distributed roughly normally, however their variances are obviously not equal. Therefore, we will conduct a right-tailed Welch t-test to check the hypothesis, setting alpha=0.05.

```
alpha=0.05
dem_counties_mhi=df_raw[df_raw.county_dem_winner==True].median_household_income.values
rep_counties_mhi=df_raw[df_raw.county_dem_winner==False].median_household_income.values
t_value, p_value = stats.ttest_ind(dem_counties_mhi, rep_counties_mhi, equal_var = False, alternative='g

if p_value <alpha:
    print("Conclusion: since p_value {: .10f} is less than alpha {} ". format (p_value ,alpha))
    print("Reject the null hypothesis that Median household income values in counties won by democrats a

else:
    print("Conclusion: since p_value {: .10f} is greater than alpha {} ". format (p_value ,alpha))
    print("Fail to reject the null hypothesis that Median household income values in counties won by dem
```

```
Conclusion: since p_value  0.0000000000 is less than alpha 0.05
Reject the null hypothesis that Median household income values in counties won by democrats are less or
equal than in counties won by republicans.
```

As we can see, the p-value is very small, therefore, the null is rejected, and we can confidently say that median household income values in counties won by democrats are higher than in counties won by republicans. Obviously, median household income is not the main parameter predicting the outcome of an election in a county, but it is definitely a parameter to consider.

7. Suggestions for next steps in analyzing this data

The data analyzed did not include counties in Utah. They could be included to increase the sample and focus on republican vote share.

More could be done to analyze third-party vote share, perhaps excluding states with no third party on the ballot, such as California.

There could be similar analyses done on subsets of states won by Democrats or Republicans, or on a subset of 'swing' states where the vote difference was smaller than a chosen threshold.

8. A PARAGRAPH THAT SUMMARIZES THE QUALITY OF THIS DATA SET AND A REQUEST FOR ADDITIONAL DATA IF NEEDED

The data quality in the senate dataset was atrocious. There were no missing data, but it seemed like in every state the data was organized somewhat differently. Obviously, this is caused by differences in state election laws, but still, the dataset was quite raw and it took long time to conduct data cleaning. The economic dataset was missing data for one county in South Dakota.

In future, I would like to get access to similar data from other elections to further study the correlations that were found.