Team Members:
Ngozi Omatu
Song Xie
Matan Ziegel
Gil Lotzky
Anna Xie

Baystate Banner: LatinX Republican Support

Project Description:

The Bay State Banner, an independent newspaper based in Boston Massachusetts, would like to understand the components of support for Republicans over the period of 2014-2020 which included two Presidential elections and two Governor's races. The goal of the project is to find whether or not there is a significant difference in the voting pattern of the LatinX community from 2014 to 2020. We started by collecting data from cities with a majority LatinX population and non-LatinX populations. We then compared the two populations to see whether there is a significant difference in voting patterns between the LatinX community and the control group (which has voted Republican consistently in the past). This analysis is then furthered into comparing the changes in LatinX voting patterns between elections within 2014-2020 for all cities in Massachusetts.The main goal is to conclude which LatinX voters changed their votes and which voters stayed consistent.

Executive Summary:

The goal of this project was to analyze whether or not there was a significant difference in LatinX voting patterns in Massachusetts from 2016-2020. Due to the unavailability of polling data, we were only able to conduct our analyses using election results and demographic data. Our team analyzed whether the change in the LatinX demographic within each tract of Massachusetts had a significant effect (in terms of R^2 * for the linear regression model) on changes in election results, specifically Republican R^2 trend line indicates the changes in LatinX population with the changes in Repulican support. The main challenge with this approach is that we can only infer whether changes in LatinX demographics affected voting patterns based on the degree of correlation. However, just because two datasets are correlated does not imply that one variable caused the changes in the other. Our analysis indicated no significant correlation between changes in LatinX demographics and changes in election results which led us to infer that there is no correlation between the change in the LatinX populations and the change in Republican support.

*R^2 measures how good of a fit a model is for a given dataset. The higher the value, the more likely that the data is explained by the model. I.e. R^2 value of 0.5 or higher for an upward trend of LatinX Republican support indicates a good chance that the data truly follows that trend

Terms of Reference:

Our client, the Baystate Banner newspaper organization, wanted to understand the components of the increase in support for Republicans throughout 2014-2020 which includes the Presidential elections and the Governor's races. Our task is to analyze whether or not there was a significant difference in LatinX voting patterns in Massachusetts from 2016-2020. We were given three key questions to answer:

1. How has support for Trump shifted across the LatinX population from 2016 to 2020? Was there a significant shift?

2. What is the breakdown of LatinX sub-groups in their support for Democratic vs. Republican candidates?

3. Which LatinX groups exhibited changes in their votes and which groups remained the same?

We were originally given six cities (Lawrence, Lynn, Springfield, Southwick, Acushnet, and Douglas) before expanding to all the cities in Massachusetts. This project was authorized by Boston University Spark, a learning organization for students at BU to lead computational and data-driven projects while gaining the experience and network for future careers. This project started in January 2021 and was completed in April 2021.

Data Resources:

- PD43 MA state portal
  - 2016 and 2020 Presidential Elections
  - 2014 and 2018 MA Governor (General Elections)
- Census data (Demographics)
  - 6 cities Census data 2019
  - Lawrence: 2016 and 2019
  - Lynn: 2016 and 2019
  - Springfield: 2016 and 2019
  - Southwick: 2016 and 2019
  - Acushnet: 2016 and 2019
  - Douglas: 2016 and 2019

    Optional Support data
- LatinX Origin Demographics (2014)

Data processing:

2014 & 2018 Mass. Demographic dataset+Election Results
Scattorpoints of the shift in Democrats and Republican votes for Presidential
Bar Data for the Government Demographic
Cleaned Datasets

Methods:

We gathered LatinX demographics data from the US government's census website and election results data from Massachusetts' election results website. Our focus from the last deliverable was to encompass both the shifts in the overall LatinX population and shifts within each sub-group and compare them to the shifts in the overall election results for both Presidential (2016 - 2020) and Governor (2014 - 2018) elections. We increased our sample size from the original six cities to all cities in Massachusetts to help us better infer a correlation between demographic changes and election results. We used linear regression models to analyze the percentage point changes between support for each political party and the percentage point changes in the LatinX population and its subgroups.

Tract Data:

Cleaning of Data: The data used in this deliverable consisted of the population data for all tracts in Massachusetts for 2014, 2016, 2018, and 2019. Some basic deletion of unnecessary rows was performed in excel. The data was read into DataFrames, 2-dimensional spreadsheets, by using the read_csv() function from the Pandas library. We separated the important columns from the main DataFrame into a separate DataFrame for processing. These columns consisted of the population data attributes of each LatinX subgroup, the general population count, and the tract number. The columns were then renamed for simpler understanding and the tracts were converted from float values to string values, as they are identifiers and not numerical values. We then merged the tract DataFrames for 2014 and 2018 as well as the DataFrames for 2016 and 2019. This made it easier to merge the comprehensive population data to their respective election data.

New columns were created to describe the percentage changes for the LatinX populations. Additionally, new latitude and longitude columns with arbitrary values were added to help convert the DataFrame to a geoPandas DataFrame. GeoPandas is an open source library that allows you to work with geospatial data. We then read in the Massachusetts tract shapefile using the geoPandas read_file function and converted the tracts from floats to strings to directly match the values from the population data with the shapefile. This conversion is a necessary step as the DataFrames will not merge correctly if they do not find an exact match for their respective tracts.

Using geoPandas merge, we can combine our population data with our geospatial tract data. This is important to correctly map population data and merge it with election data. We then dropped all unnecessary attributes and renamed columns for further processing with geoPandas. This completes the necessary processing for the tract and population data.

Precinct Data:

We first read in the 2016 and 2020 Presidential election data as well as the 2014 and 2018 Governor's election data using the Pandas read_csv function. We also read the precinct shapefile for Massachusetts into a geoPandas DataFrame. We then renamed columns to be consistent with categorical names (Democratic and Republican). We dropped all empty cells using the dropna() function. However, we encountered an error when trying to combine the DataFrames, so it was important to diagnose the problem. No precincts with missing wards appeared after merging the data so we attempted to look closely at those cases. After setting boolean statements for five sample cells, it was showing the wards were False when set equal to each other. After printing out a sample from each set, the Presidential data for 2020 had a space in front of it that was difficult to notice without printing the data set. The spaces were stripped and we were now ready to merge the Presidential data with the geospatial DataFrame for precincts.

The matching Presidential data and Governor's data were successfully merged into a single Pandas DataFrame. We then appended arbitrary latitude and longitude columns to the merged DataFrame to convert it to a geoPandas DataFrame. We dropped the unnecessary columns from the geoPandas DataFrame and set the city/town values to uppercase to match the cases with the geospatial DataFrame for the precincts. We replaced the None values with '-' to create exact match cases. We then successfully merged all the Presidential election and Governor's election data with their respective precinct geospatial DataFrames. Further cleaning was done by renaming the attributes and dropping unnecessary information.

Final Dataset:

We created two geoPandas DataFrames containing the Presidential and Governor's voting information with their respective geospatial information. We also created DataFrames for the population information and their respective geospatial information. This allowed us to merge all the DataFrames using a spatial join. A spatial join was performed on points of intersection. The data sets were then converted into csv files where further analysis can be performed. No further issues with the DataFrames were present.

Regression Cleaning:

We took the .CSV output from Final Dataset and turned it into a DataFrame object using the Pandas read_csv() function for further processing. Every value in each column of the DataFrame is currently a string which allows us to remove any substring that prevents us from converting each value to either an integer or float such as unnecessary spaces, NaN, and commas using .replace(). After cleaning, we converted the values of the necessary columns (any column containing voting data and demographic data) to integers/floats using .to_numeric(). This allowed us to conduct mathematical computations on the values of the desired columns.

Regression:

We calculated the percentage point change over a 4-year interval using these methods:

Percentage Point Change in Democratic Support:
(Dem votes 2020 / total votes 2020) - (Dem votes 2016 / total votes 2016)

Percentage Point Change in Republican Support:
(Rep votes 2020 / total votes 2020) - (Rep votes 2016 / total votes 2016)

Percentage Point Change in LatinX subgroups:
(subgroup pop 2020 / total pop 2020) - (subgroup pop 2016 / total pop 2016)

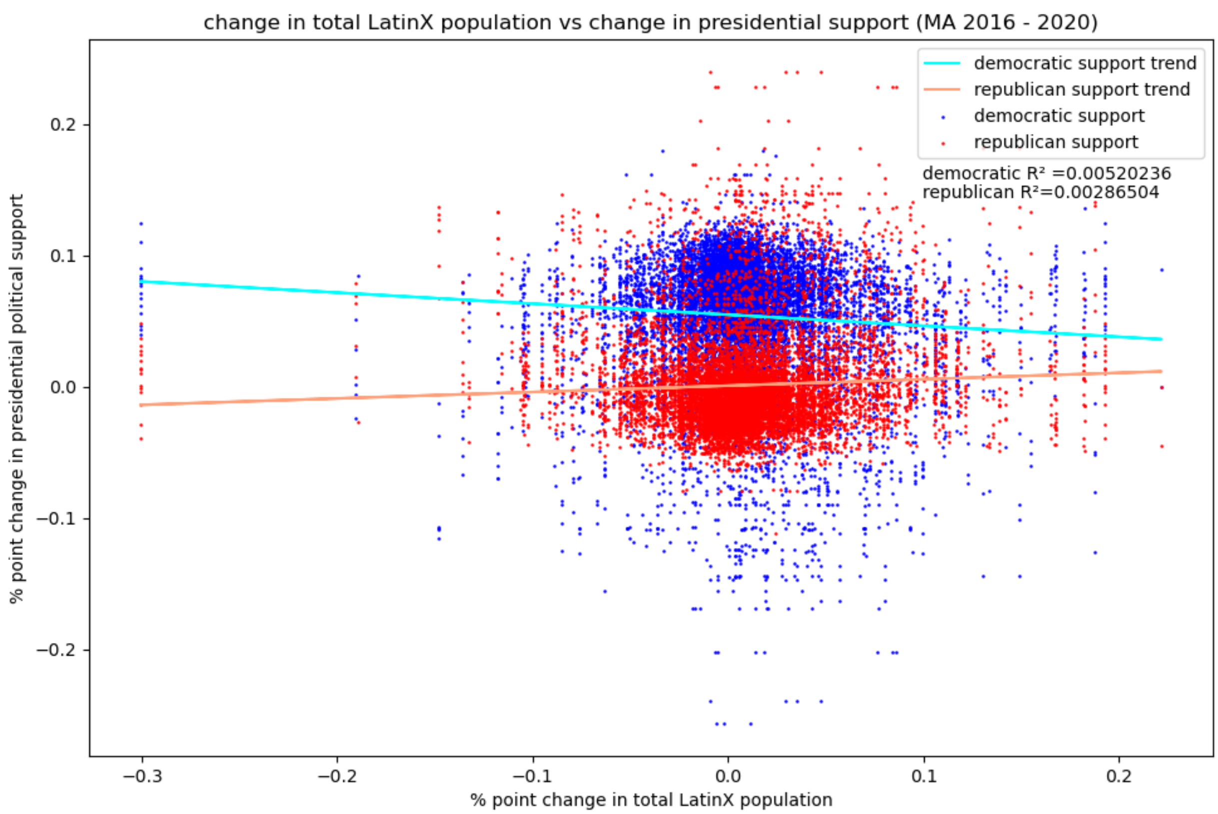
Percentage Point Change in Total LatinX Pop:
(sum of subgroup pop 2020 / total pop 2020) - (sum of subgroup pop 2016 / total pop 2016)

After all of the calculations, we created two separate DataFrame objects to hold only the Presidential election and demographic data and Governor's election and demographic data. Then we created a scatter plot using matplotlib's .scatter() for a pair of DataFrame columns (percentage point change in population, percentage point change in political support) which includes data for all tracts in Massachusetts.

We then fitted the same columns that were used for the scatter plot to Sklearn's LinearRegression package using LinearRegression().fit() and then plotted the predicted line of best fit onto the same scatter plot. This was done for the LatinX total population and its subgroups and each political party. We also added each line of best fit's R^2 value using sklearn's r2_score package to identify the significance of the correlation between the percentage point change of LatinX population and the percentage point change of the support for each political party.

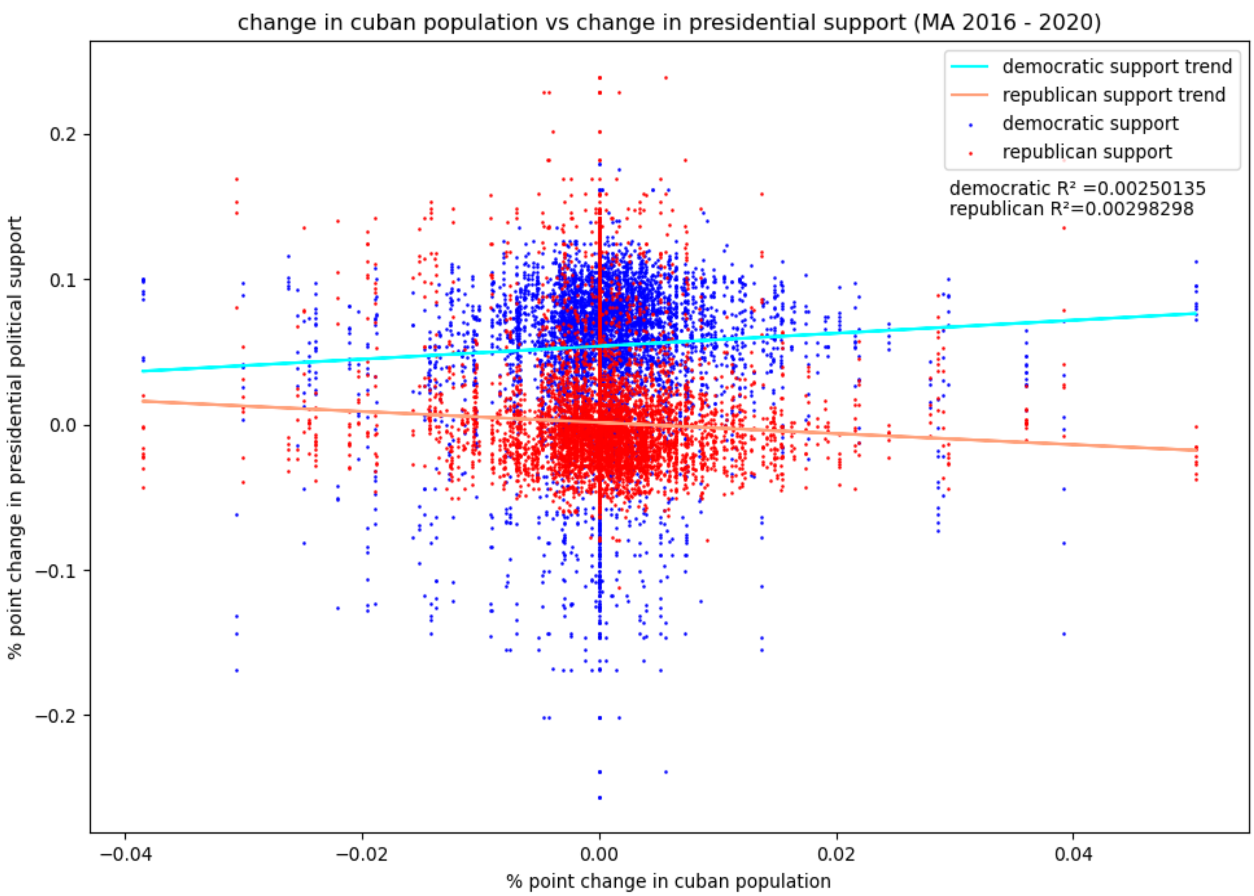
Results:
Visualizations:

Graph 1: LatinX Population vs. Political Support (Presidential Election) Changes (2016-2020)



Democratic Support: R-squared is ~0, therefore, we cannot conclude that the changes in Democratic support from 2016 to 2020 can be explained by the changes in the LatinX population.
Republican Support: R-squared is ~0, therefore, we cannot conclude that the changes in Republican support from 2016 to 2020 can be explained by the changes in the LatinX population.

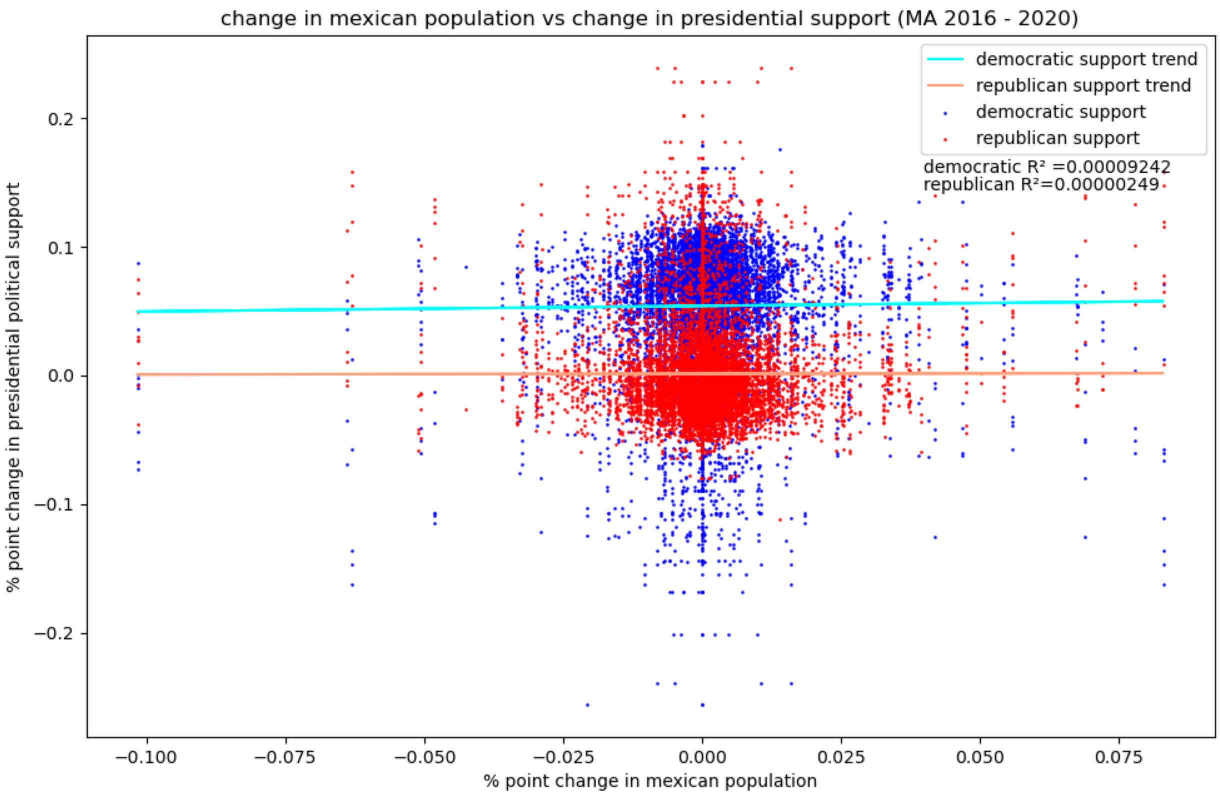
Graph 2: Presidential Support v. Cuban Population Change (2016-2020)



Democratic Support: R-squared is ~0, therefore, we cannot conclude that the changes in Democratic support from 2016 to 2020 can be explained by the changes in the Cuban population.

Republican Support: R-squared is ~0, therefore, we cannot conclude that the changes in Republican support from 2016 to 2020 can be explained by the changes in the Cuban population.

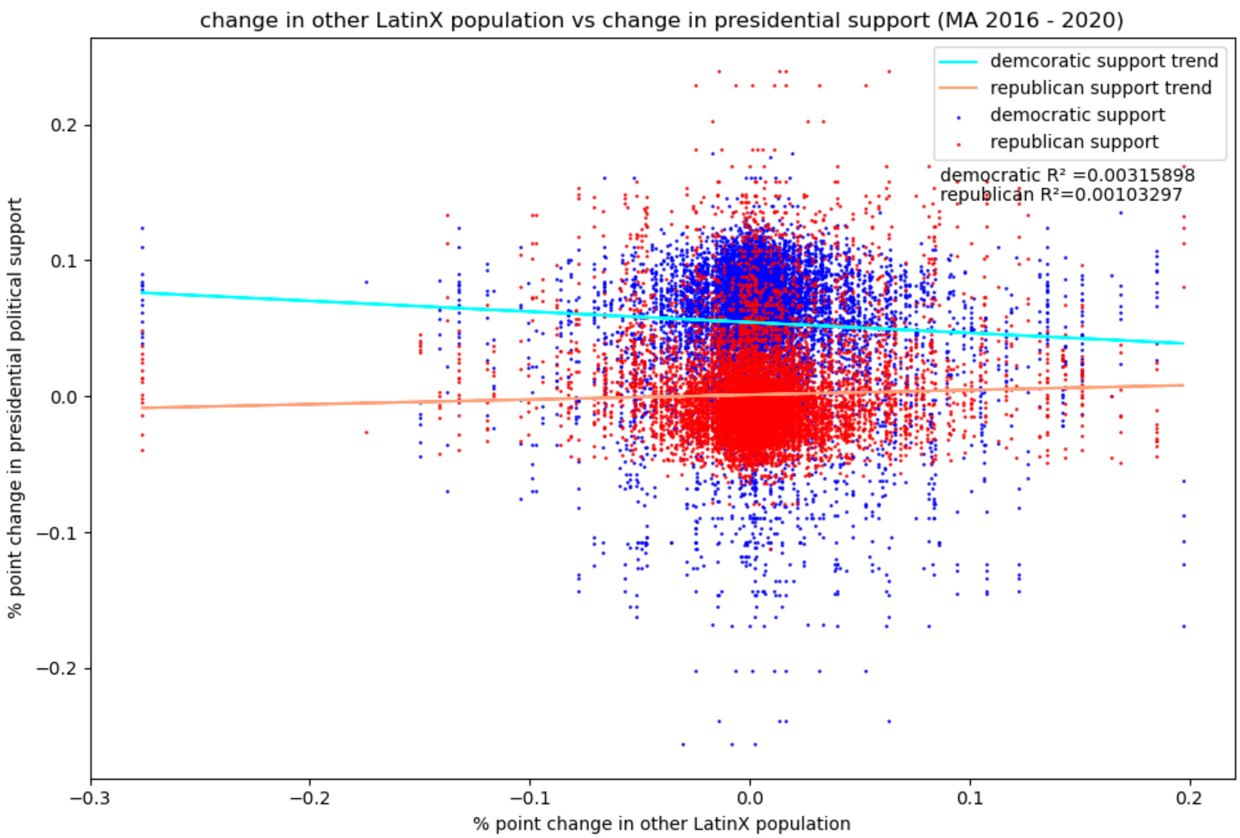
Graph 3: Presidential Support v. Mexican Population Change (2016-2020)



Democratic Support: R-squared is ~0, therefore, we cannot conclude that the changes in Democratic support from 2016 to 2020 can be explained by the changes in the Mexican population.

Republican Support: R-squared is ~0, therefore, we cannot conclude that the changes in Republican support from 2016 to 2020 can be explained by the changes in the Mexican population.

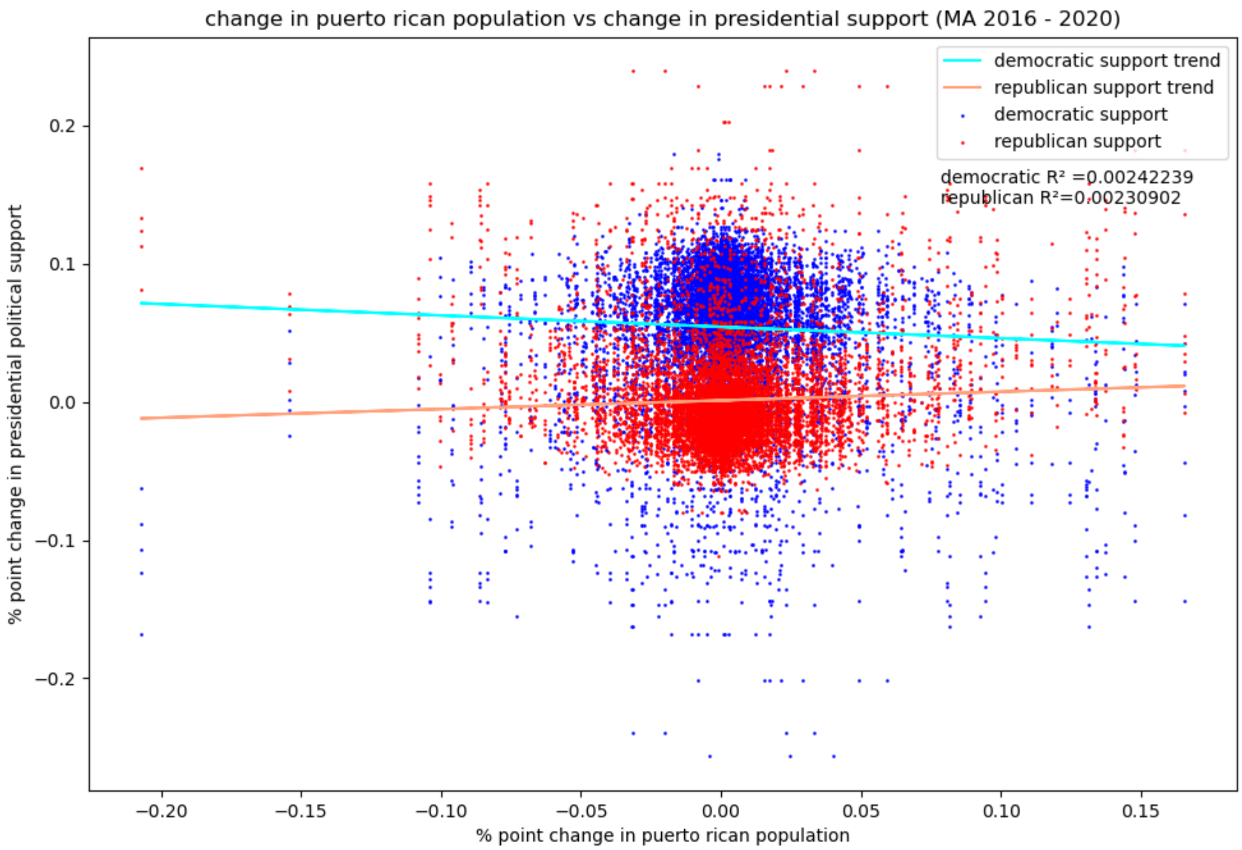
Graph 4: Presidential Support v. other LatinX Population Change (2016-2020)



Democratic Support: R-squared is ~0, therefore, we cannot conclude that the changes in Democratic support from 2016 to 2020 can be explained by the changes in Other LatinX population.

Republican Support: R-squared is ~0, therefore, we cannot conclude that the changes in Republican support from 2016 to 2020 can be explained by the changes in the Other LatinX population.

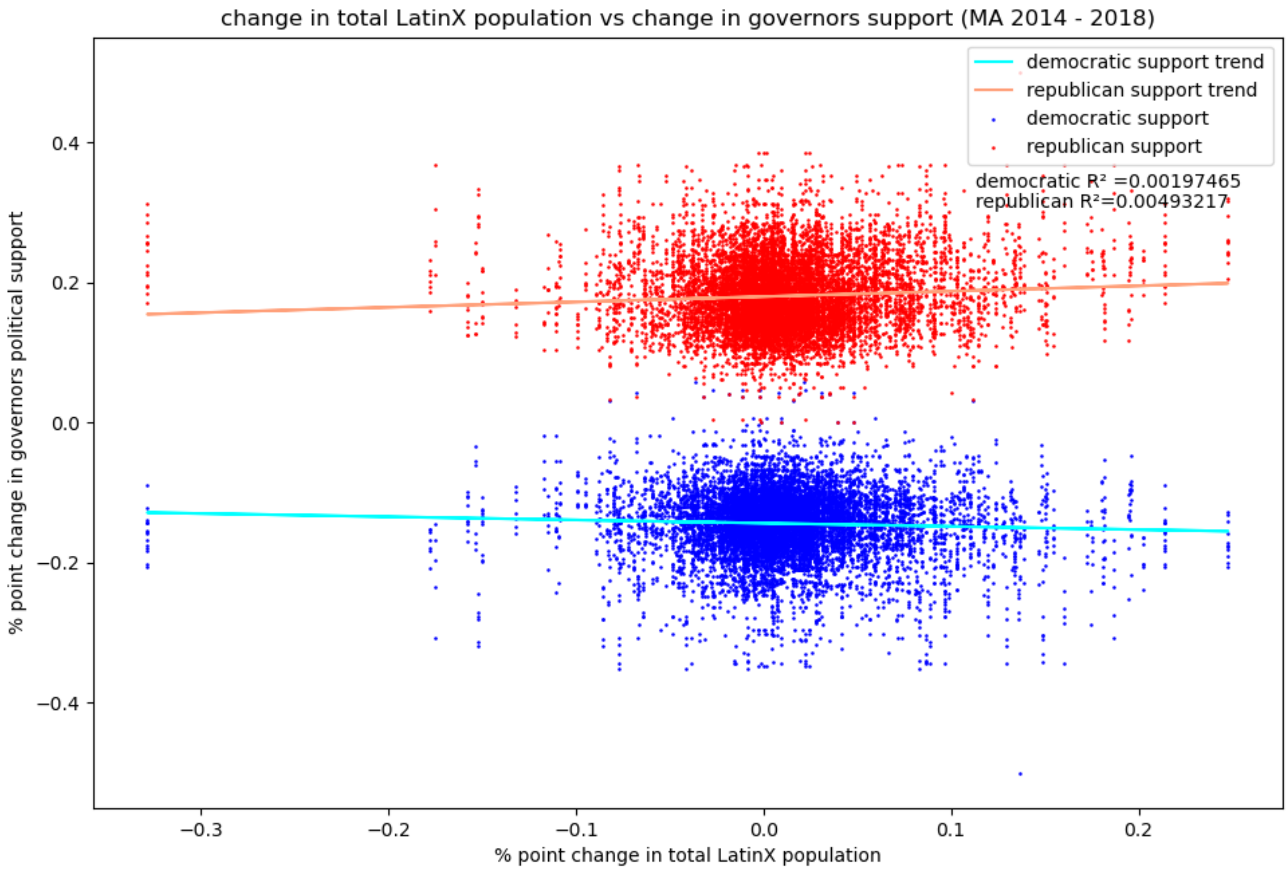
Graph 5: Presidential Support v. Puerto Rican Population Change (2016-2020)



Democratic Support: R-squared is ~0, therefore, we cannot conclude that the changes in Democratic support from 2016 to 2020 can be explained by the changes in the Puerto Rican population.

Republican Support: R-squared is ~0, therefore, we cannot conclude that the changes in Republican support from 2016 to 2020 can be explained by the changes in the Puerto Rican population.

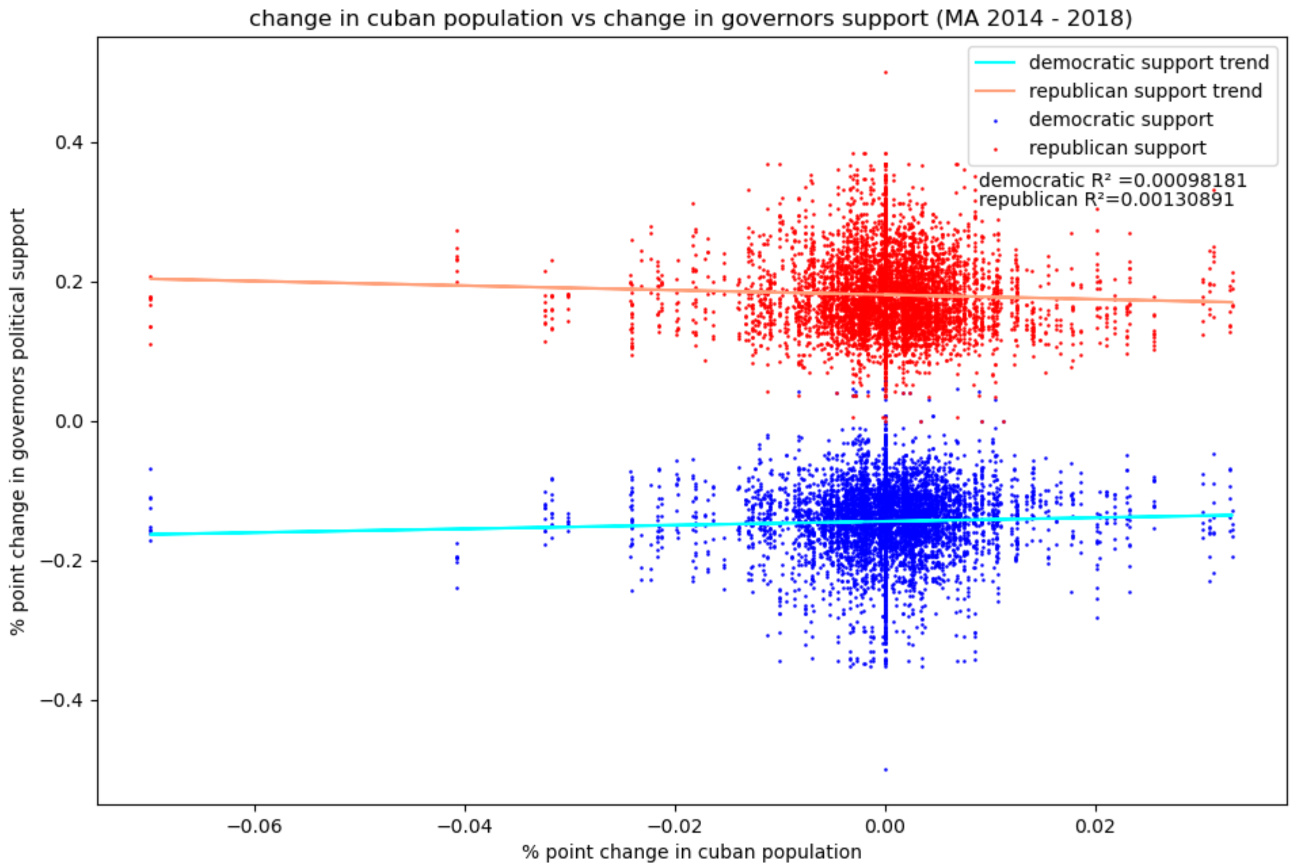
Graph 6: LatinX Population vs. Political Support (Governor's Election) Changes (2014-2018)



Democratic Support: R-squared is ~0, therefore, we cannot conclude that the changes in Democratic support from 2014 to 2018 can be explained by the changes in the LatinX population.

Republican Support: R-squared is ~0, therefore, we cannot conclude that the changes in Republican support from 2014 to 2018 can be explained by the changes in the LatinX population.

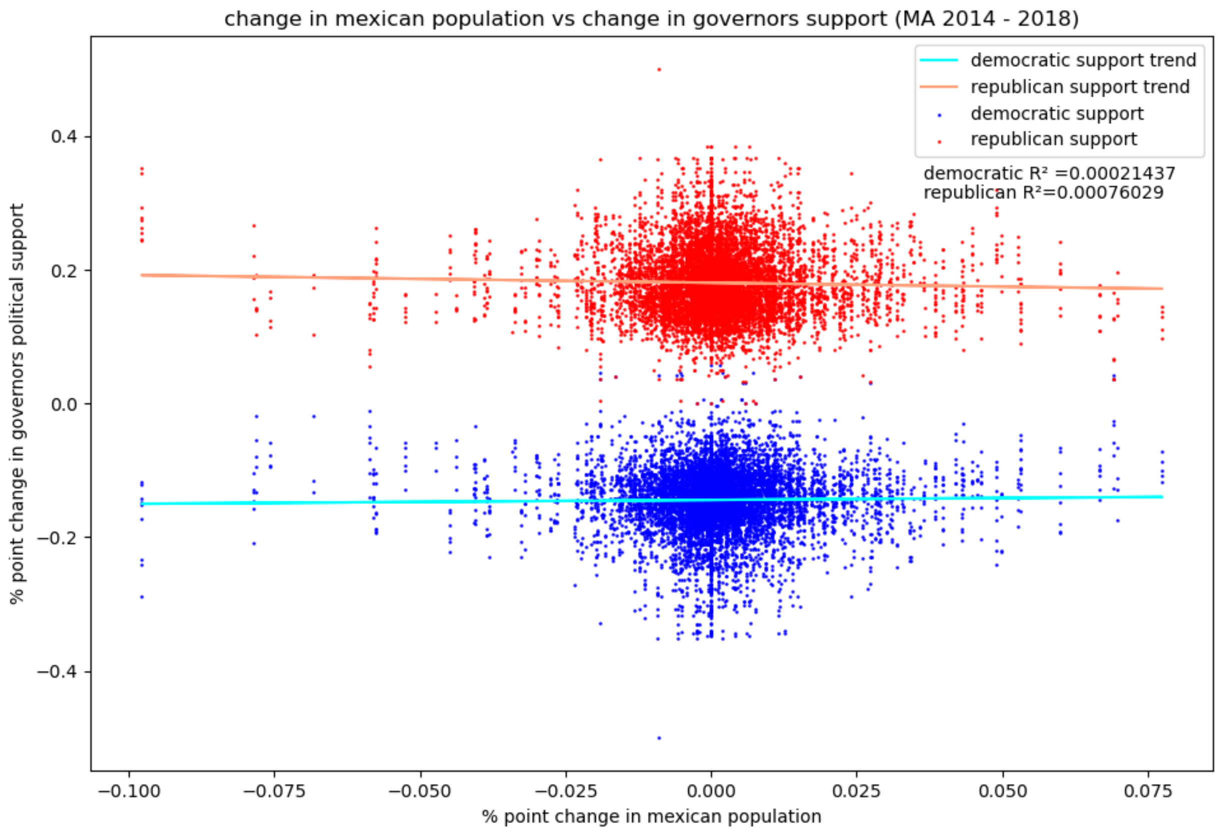
Graph 7: Governor's Support v. Cuban Population Change (2014-2018)



Democratic Support: R-squared is ~0, therefore, we cannot conclude that the changes in Democratic support from 2014 to 2018 can be explained by the changes in the Cuban population.

Republican Support: R-squared is ~0, therefore, we cannot conclude that the changes in Republican support from 2014 to 2018 can be explained by the changes in the Cuban population.

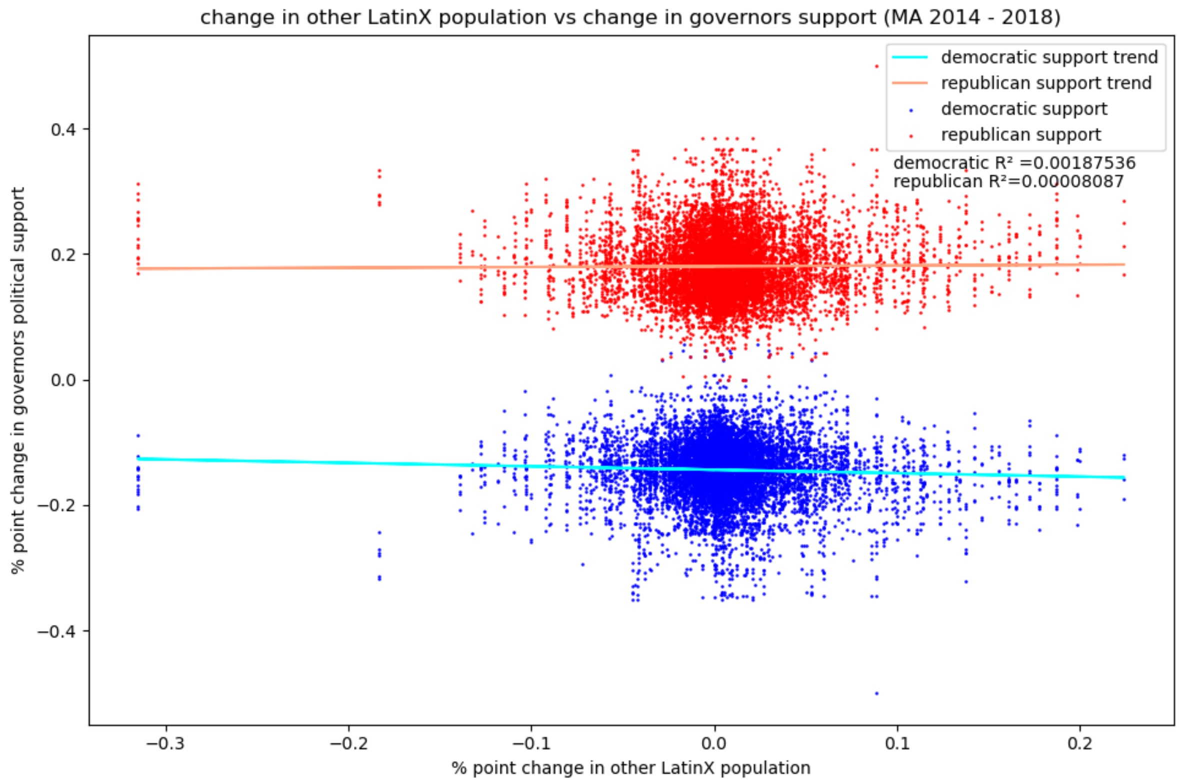
Graph 8: Governors Support v. Mexican Population Change (2014-2018)



Democratic Support: R-squared is ~0, therefore, we cannot conclude that the changes in Democratic support from 2014 to 2018 can be explained by the changes in the Mexican population.

Republican Support: R-squared is ~0, therefore, we cannot conclude that the changes in Republican support from 2014 to 2018 can be explained by the changes in the Mexican population.

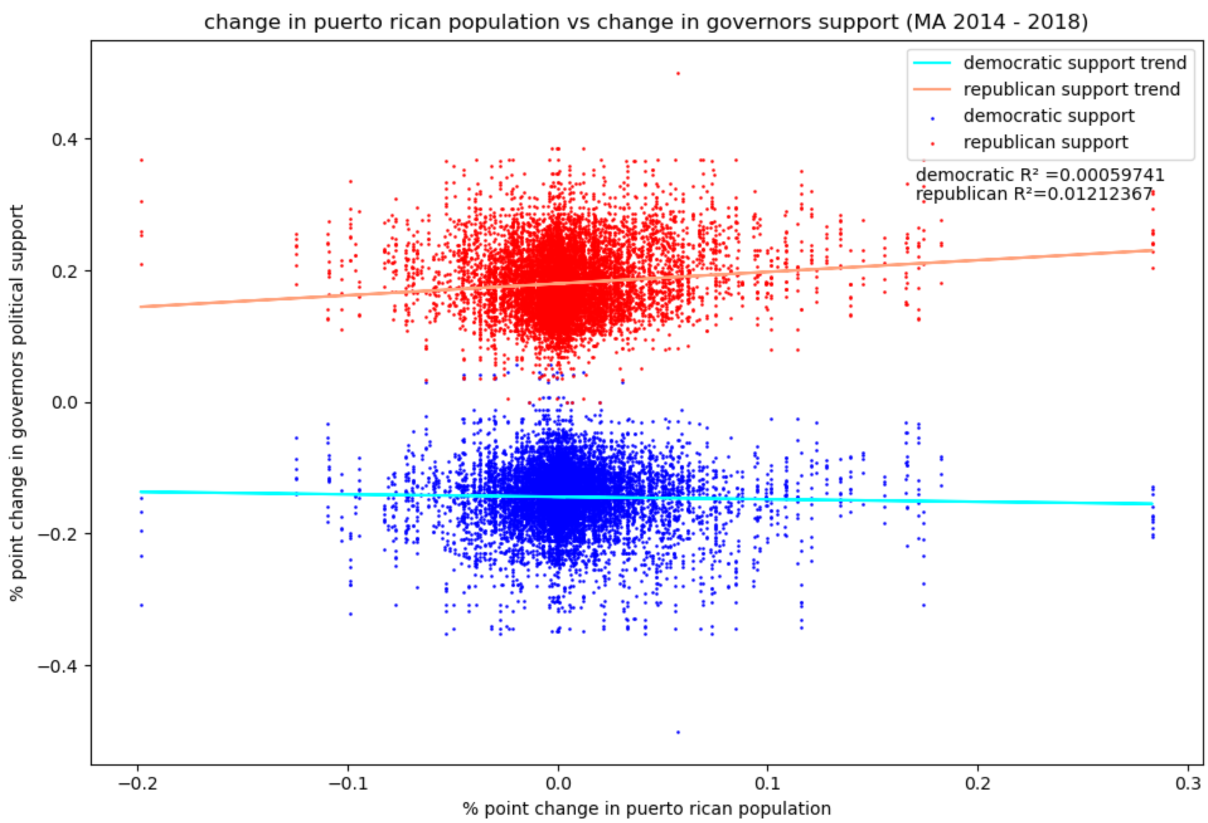
Graph 9: Governors Support v. other LatinX Population Change (2014-2018)



Democratic Support: R-squared is ~0, therefore, we cannot conclude that the changes in Democratic support from 2014 to 2018 can be explained by the changes in Other LatinX population.

Republican Support: R-squared is ~0, therefore, we cannot conclude that the changes in Republican support from 2014 to 2018 can be explained by the changes in the Other LatinX population.

Graph 10: Governors Support v. Puerto Rican Population Change (2014-2018)



Democratic Support: R-squared is ~0, therefore, we cannot conclude that the changes in Democratic support from 2014 to 2018 can be explained by the changes in the Puerto Rican population.

Republican Support: R-squared is ~0, therefore, we cannot conclude that the changes in Republican support from 2014 to 2018 can be explained by the changes in the Puerto Rican population.

Additional Visualizations:

Fig 1.

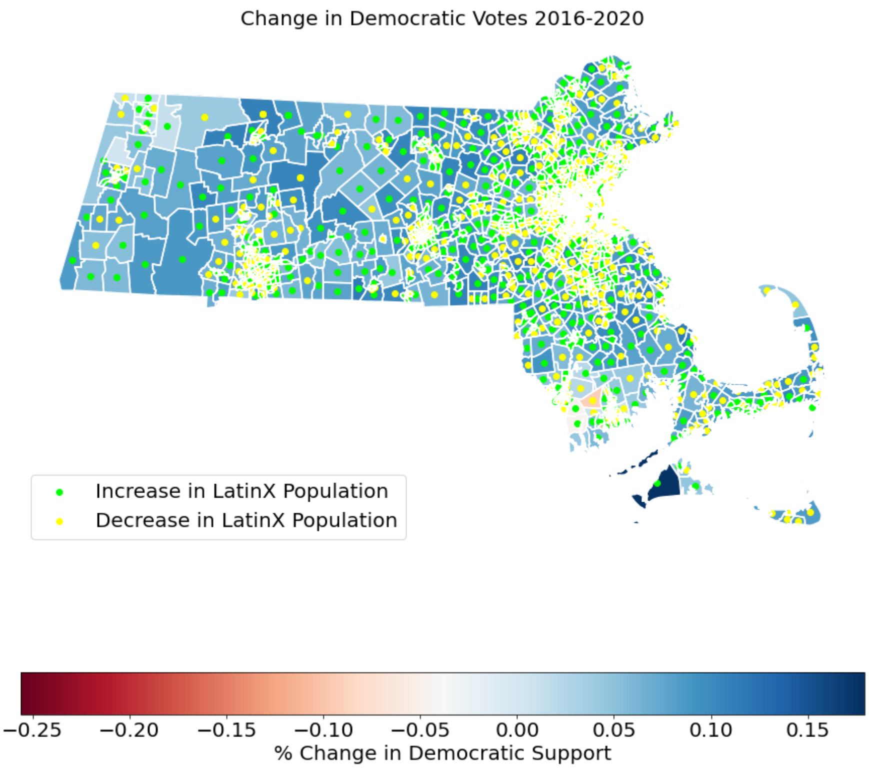


Fig 1 description: The visualization shows a heatmap of the changes in Democratic support for the 2016 to 2020 Massachusetts Presidential election. Coloration of red indicates a decrease in democratic support. Coloration of blue indicates a positive change in Democratic support. Additionally, the dots depict changes in the LatinX populations within each tract. A yellow dot shows a decrease in the LatinX population. A lime dot shows an increase in LatinX population.

Major Democratic Support changes: West Tisbury ,Chilmark, Aquannah
Major Population Changes: Hopedale, Nantucket, Dudley

Fig 2:

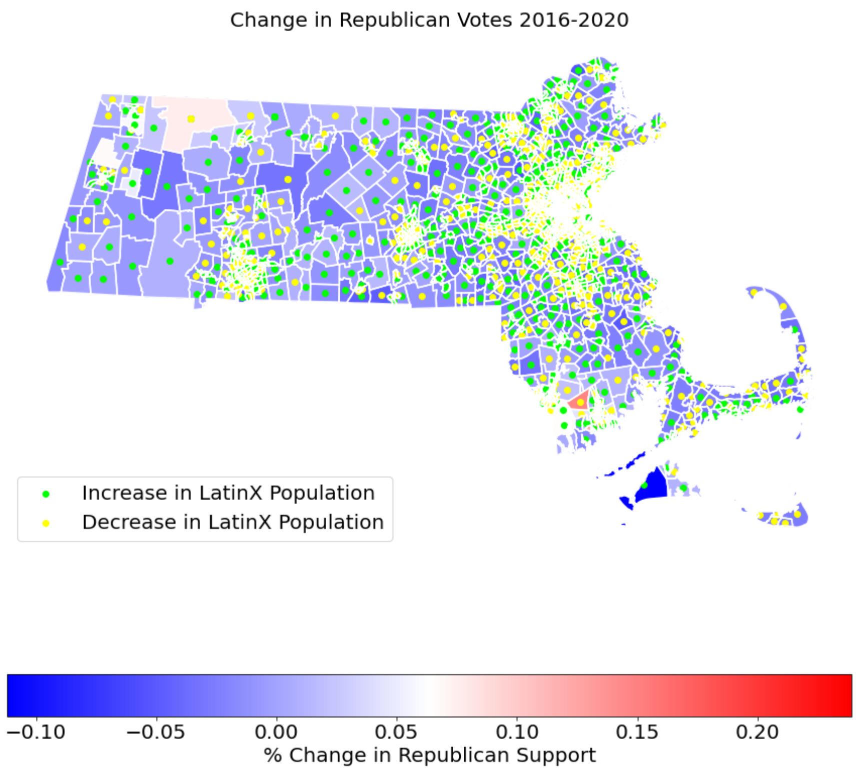


Fig 2 description: The visualization shows a heatmap of the changes in Republican support for the 2016 to 2020 Massachusetts Presidential election. Coloration of red indicates a decrease in Republican support. Coloration of blue indicates a positive change in Republican support. Additionally, the dots depict changes in the LatinX populations within each tract. A yellow dot shows a decrease in the LatinX population. A lime dot shows an increase in LatinX population.

Major Support changes: West Tisbury ,Chilmark, Aquannah
Major Population Changes: Hopedale, Nantucket, Dudley

Fig 3:

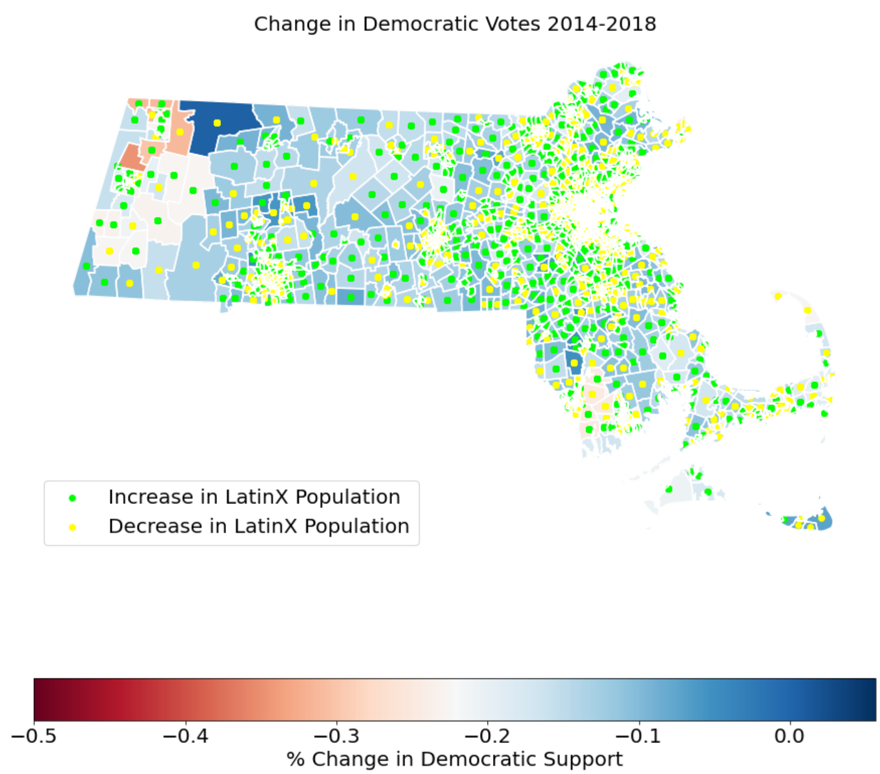


Fig 3 description: The visualization shows a heatmap of the changes in Democratic support for the 2014 to 2018 Massachusetts Governor election.Coloration of red indicates a decrease in Republican support. Coloration of blue indicates a positive change in Democratic support. Additionally, the dots depict changes in the LatinX populations within each tract. A yellow dot shows a decrease in the LatinX population. A lime dot shows an increase in LatinX population.

Major support changes: Lanesborough ,Rowe ,Colrain
Major Population changes: Hopedale, Athol, Avon

Fig 4:

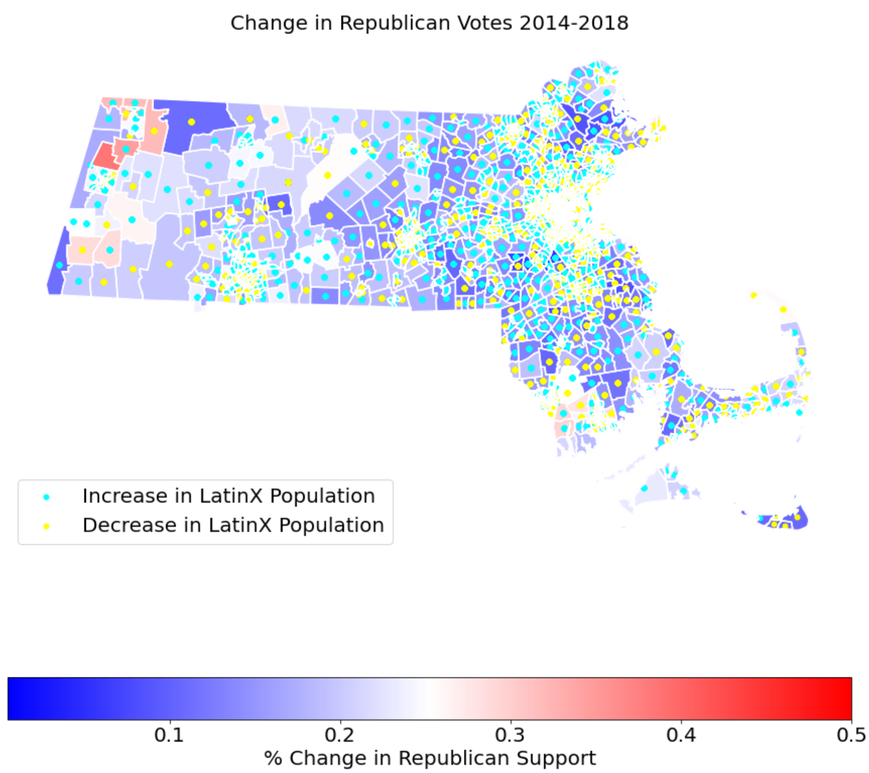


Fig 4 description: The visualization shows a heatmap of the changes in Republican support for the 2014 to 2018 Massachusetts Governor election. Coloration of red indicates a decrease in Republican support. Coloration of blue indicates a positive change in Republican support. Additionally, the dots depict changes in the LatinX populations within each tract. A yellow dot shows a decrease in the LatinX population. A lime dot shows an increase in LatinX population.

Major support changes: Lanesborough, Alford, Egremont **Major Population changes**: Hopedale, Athol, Avon

Conclusion:

Our analysis showed no significant correlation between changes in the LatinX population and changes in support for a political party which led us to believe that Republican support from the LatinX community did not significantly shift from 2016-2020. However, this conclusion contradicts some research from external sources which led us to believe that there may have been flaws in our analyses. One source of the problem may have been the fact that we conducted analyses on the changes in the LatinX population in regards to the changes in the total number of votes for each tract. What we should have analyzed were the changes in the LatinX population in regards to the changes in the total votes that only came from LatinX individuals. However, this data was not publicly available to us which may have caused the difference in conclusions.

Key Question 1:

Our team analyzed changes in both the LatinX population and Presidential election support within each Tract in Massachusetts. We looked at how changes in the total LatinX population correlate with changes in the Presidential election results from 2016 to 2020 in Graph 1 using linear regression.

For the changes in Republican support relative to the changes in the total LatinX population, the trend line is almost flat which indicates that changes in LatinX population had little to no effect on the changes in Republican support. In other words, changes in the LatinX population had no significant impact on the election outcome.

This is also supported by the trend line's R^2 value of 0.0029.This implies that almost none of the variance in the change in Republican support can be explained by the changes in the LatinX population. Therefore, we cannot conclude that changes in the total LatinX population had an effect on changes in Republican support from 2016 to 2020.
The geospatial mapping of Fig 1 (changes in Republican votes and changes in LatinX Population) showed that 23.85% of tracts in Massachusetts that experienced an increase in LatinX population saw an increase in Republican Support. In addition, 36.60% of tracts in Massachusetts experienced an increase in LatinX population and a decrease in Republican Votes, 24.16% of tracts experienced a decrease in LatinX population and a decrease in

Republican Votes, and 14.57% of tracts in Massachusetts experienced a decrease in LatinX population and an increase in Republican Support. This also supports our inference that changes in the LatinX population had no significant effect on changes in Republican support.

Key Question 2:

None of the LatinX subgroups in Massachusetts show significant support towards one specific Presidential party. The linear regression models showed that the variation in Presidential support from 2016 to 2020 is not explained by the change in any of the subgroup populations. Since we did not have access to election data organized by subgroups, our conclusion is derived from the correlation between the changes in the LatinX subgroup populations and changes in Democratic and Republican support (in terms of votes). Even if there appears to be a correlation between changes in the LatinX subgroup population and changes in political support for a specific political party, we must recognize that correlation does not imply causation.

Key Question 3:

We had difficulty answering this question because we did not have access to data that showed how each LatinX subgroup had voted for each election. To see how the LatinX subgroup voting patterns changed, we needed data regarding how these subgroups had voted (i.e. who they voted for). Since this data was not available, we were limited in our ability to analyze changes in LatinX voting patterns and how they may have changed from 2016 to 2020.

The data that we analyzed included changes in the population of LatinX subgroups in Massachusetts as well as changes in overall election results. It would not be practical to infer changes in LatinX voting patterns purely based on changes in the LatinX demographics and election results for all demographics rather than just election results for LatinX demographics. The correlation between changes in LatinX subgroups and changes in election results do not yield any insight into the actual voting patterns of those subgroups. Therefore, our team was limited in being able to analyze the change in the voting patterns of LatinX subgroups.

Limits & Challenges:

A major limitation of this project was that our data did not go into detail about the demographic breakdown of votes, specifically within the LatinX population. The lack of connection between city election results and LatinX demographics data made it difficult to directly answer the key questions for this project. Therefore, we had to resort to analyzing LatinX voting patterns by comparing changes in the LatinX population relative to changes in the election results of all demographics. This method was not ideal because we cannot assume that changes in the LatinX demographics had any relation to election results. There could be several external factors that can cause shifts in election results that are not at all related to changes in the LatinX population.

Next Step to Improve:

Having access to voting data broken down by demographics would help improve our analysis. With our current data, we were only able to analyze the correlation between changes in the LatinX population relative to changes in total votes for all demographic groups. However, it would be much more effective to analyze changes in the LatinX population relative to changes in the total votes that only came from LatinX individuals.

A potential new project could be to see which LatinX cities supported Trump and to find the contributing factors for their support. There is the possibility that the voter turnout information for the LatinX population in the 2020 Presidential election will soon be available. This data would allow us to properly analyze the LatinX voter population and compare our results to that of other racial groups. We could also analyze changes in voter patterns based on gender and see if there is any relationship between gender and political support. We can also compare the pre-election polling data from 2016 - 2020 to the election results in order to assess the accuracy of the pre-election data.

Research:

Our team conducted analyses in order to gain a better sense of the potential connection between changes in the LatinX population to changes in political support in Massachusetts. 73.8% of towns saw an increase in the LatinX population from 2016-2019, while only 24.3% of towns in Massachusetts saw an increase in Republican support from 2016-2020. However, the five towns with the highest LatinX percentage in 2019 all showed a percentage point increase in Republican support from 2016-2020. All 5 towns with the highest concentration of LatinX population showed significant increases in Republican support. These results directly correlated to our external research, which emphasized the fact that Trump built Latino support in a number of gateway cities[1] in Massachusetts. Although Joe Biden got almost three times as many votes as Donald Trump in Lawrence, Trump's share still increased from 14% in 2016 to 25% in 2020[1].

---

[1] Gateway cities are midsize urban centers that anchor regional economies around the state. For generations, these communities were home to industry that offered residents good jobs and a "gateway" to the American Dream.[3]

City/Town	% LatinX 2019	Democratic Percentage point Change	Republican Percentage point Change
LAWRENCE	0.747413	-10.425567	12.561895
CHELSEA	0.579951	3.134887	4.054185
HOLYOKE	0.405884	-1.616373	6.618964
SPRINGFIELD	0.400684	-3.071012	4.796808
METHUEN	0.397335	4.121485	0.773999

The increase in LatinX support for Trump in Massachusetts can be due to several factors. For example, many members of the LatinX population were positively influenced by the coronavirus stimulus package. The stimulus check influenced many members of the LatinX population who were undecided on who to vote for[1]. In addition, much of the LatinX population in MA could have been influenced by the increase of anti-communist propaganda against democrats[1]. This most likely had the strongest influence on Cuban and Venezuelan immigrants, who had experienced socialism in the past. The increase in LatinX support in these towns could also be attributed to Trump's anti-abortion stance[2]. Evangelical LatinX voters in MA and first-generation citizens from very Catholic countries were in support of Trump's anti-abortion stance and therefore might have been more inclined to vote for him.

References:

[1] Zea, Tibisay, and Simón Rios. “Trump Lost In Massachusetts, But Built Latino Support In Gateway Cities.” Trump Lost In Massachusetts, But Built Latino Support In Gateway Cities | WBUR News. WBUR, November 16, 2020. https://www.wbur.org/news/2020/11/16/latino-vote-massachusetts-gateway-cities-trump.

[2] Jonas, Michael. “One Place Trump Gained in Mass.: Heavily Latino Cities .” CommonWealth Magazine, November 6, 2020. https://commonwealthmagazine.org/politics/one-place-trump-gained-in-mass-heavily-latino-cities/.

[3] “About the Gateway Cities.” MassINC. Accessed April 16, 2021. https://massinc.org/our-work/policy-center/gateway-cities/about-the-gateway-cities/.

Appendix:

GitHub repository with all python/csv files and code:

https://github.com/glotzky/LatinXBaystate.git

```
In [63]: test16['Ward']==test20['Ward']

Out[63]: 0    False
         1    False
         2    False
         3    False
         4    False
         Name: Ward, dtype: bool

In [64]: print(test20['Ward'][0])
         print(test16['Ward'][0])

          _
         _

In [65]: pres2016['Ward'] = pres2016['Ward']#.str.replace('-','0')
         pres2020['Ward'] = pres2020['Ward'].str.strip(' ') #.str.replace('-',)
```

Debugging the Ward issue

```
type(mergedP)

pandas.core.frame.DataFrame

mergedP['Latitude'] = -40.266666
mergedP['Longitude'] = 72.3452

mergedPG = gpd.GeoDataFrame(
   mergedP, geometry=gpd.points_from_xy(mergedP.Longitude, mergedP.Latitude))

mergedPG = mergedPG.drop(columns = ["Latitude","Longitude"])

type(mergedPG)

geopandas.geodataframe.GeoDataFrame
```

Converting Pandas DataFrameDataFrames to geoPandas DataFrames with arbitrary latitude and longitude columns

```
MA_t = gpd.read_file("CENSUS2010_BLK_BG_TRCT_SHP/CENSUS2010TRACTS_POLY.shp")
```

```
MA_p = gpd.read_file("wardsprecincts_poly/WARDSPRECINCTS_POLY.shp")
```

Reading in shapefile

```
FinalData.columns
```

```
Index(['STATEFP10', 'COUNTYFP10', 'TRACTCE10', 'GEOID10', 'NAME10',
       'NAMELSAD10', 'MTFCC10', 'ALAND10', 'AWATER10', 'INTPTLAT10',
       'INTPTLON10', 'AREA_SQFT', 'AREA_ACRES_left', 'POP100_RE', 'HU100_RE',
       'LOGPL94171', 'LOGSF1', 'LOGACS0610', 'LOGSF1C', 'SHAPE_AREA_left',
       'SHAPE_LEN_left', 'geometry', 'Tract', '% Point Change in LatinX Pop.',
       '% Point Change in Total Pop.', '% Point Puerto Rican Change',
       '% Point Mexican Change', '% Point Cuban Change',
       '% Point Other LatinX Change', 'Total Population 2016',
       'LatinX Population 2016', 'Mexican 2016', 'Puerto Rican 2016',
       'Cuban 2016', 'Other LatinX 2016', 'Total Population 2019',
       'LatinX Population 2019', 'Mexican 2019', 'Puerto Rican 2019',
       'Cuban 2019', 'Other LatinX 2019', 'index_right', 'WP_NAME', 'WARD',
       'PRECINCT', 'DISTRICT', 'POP_2010', 'TOWN', 'TOWN_ID', 'AREA_SQMI',
       'AREA_ACRES_right', 'YEAR', 'SHAPE_AREA_right', 'SHAPE_LEN_right',
       'City/Town', 'Pct', 'Ward', 'Democratic 2016', 'Republican 2016',
       'Total Votes Cast', 'Democratic 2020', 'Republican 2020',
       'Total Votes Cast 2020', '% Point Change in Democratic Votes',
       '% Change in Total Votes', '% Point Change in Republican Votes'],
      dtype='object')
```

```
FinalData.shape
```

```
(10567, 66)
```

Successful Merged Dataset with all columns necessary for geospatial mapping and statistical analyses.

Code:

Import Necessary Libraries

```
import pandas as pd
import numpy as np
import shapely
from shapely.geometry import shape, mapping
import geopandas as gpd
from geopandas.tools import sjoin
import requests
import json
import matplotlib.pyplot as plt
```

```
print('Necessary libraries have been imported successfully.')
```

```
Necessary libraries have been imported successfully.
```

tracts2016 = pd.read_csv('plzwrk141619/2016alltractdata.csv') tracts2019 = pd.read_csv('plzwrk141619/2019alltractdata.csv')

Load in CSV Files for Tract data(contains population data)

```
tracts2016 = pd.read_csv('plzwrk141619/2016alltractdata.csv')
tracts2019 = pd.read_csv('plzwrk141619/2019alltractdata.csv')
```

Keep necessary attribute columns only

```
proct16 = tracts2016[['Geographic Area Name','Estimate!!SEX AND AGE!!Total population','Estimate!!HISPANIC OR LATINO AND RACE!!To
```

```
proct19 = tracts2019[['Geographic Area Name','Estimate!!SEX AND AGE!!Total population','Estimate!!HISPANIC OR LATINO AND RACE!!To
```

```
proct19.columns
```

```
Index(['Geographic Area Name', 'Estimate!!SEX AND AGE!!Total population',
        'Estimate!!HISPANIC OR LATINO AND RACE!!Total population!!Hispanic or Latino (of any race)',
        'Estimate!!HISPANIC OR LATINO AND RACE!!Total population!!Hispanic or Latino (of any race)!!Mexican',
        'Estimate!!HISPANIC OR LATINO AND RACE!!Total population!!Hispanic or Latino (of any race)!!Puerto Rican',
        'Estimate!!HISPANIC OR LATINO AND RACE!!Total population!!Hispanic or Latino (of any race)!!Cuban',
        'Estimate!!HISPANIC OR LATINO AND RACE!!Total population!!Hispanic or Latino (of any race)!!Other Hispanic or Latino'],
       dtype='object')
```

Rename columns for easier processing

```
proct16 = proct16.rename(columns = {'Geographic Area Name':'Tract','Estimate!!SEX AND AGE!!Total population':'Total Population 20
proct19 = proct19.rename(columns = {'Geographic Area Name':'Tract','Estimate!!SEX AND AGE!!Total population':'Total Population 20
```

Convert Tract Number columns from float values to strings

```
proct16['Tract'] = proct16['Tract'].astype(str)
proct19['Tract'] = proct19['Tract'].astype(str)
```

Merge the two tract pandas dataframes into one that contains all the years you want to compare

```
merged = pd.merge(proct16,proct19, how = "left", left_on = ['Tract'],right_on =['Tract'])
```

```
merged
```

	Tract	Total Population 2016	LatinX Population 2016	Mexican 2016	Puerto Rican 2016	Cuban 2016	Other LatinX 2016	Total Population 2019	LatinX Population 2019	Mexican 2019	Puerto Rican 2019	Cuban 2019	Other LatinX 2019
0	101.0	2962	164	0	105	0	59	2973	124	6	23	32	63
1	102.06	3168	10	1	1	0	8	3617	44	0	0	26	18
2	102.08	1589	30	17	0	0	13	1122	17	17	0	0	0
3	103.04	2107	23	7	0	6	10	2394	20	0	0	14	6
4	103.06	2817	53	0	0	0	53	2507	23	0	0	0	23
...	...	...	...	...	...	...	...	...	...	...	...	...	...
1497	7601.0	3386	0	0	0	0	0	3441	1	0	0	0	1
1498	7611.0	5178	42	0	41	0	1	5213	62	0	0	0	62
1499	7612.0	5541	397	101	95	0	201	5506	483	154	20	0	309
1500	7613.0	3397	515	9	471	11	24	3259	341	0	161	50	130
1501	7614.0	6570	311	64	114	0	133	6569	399	68	130	24	177

1502 rows × 13 columns

```
points = merged.copy()
```

Create new columns for the changes between the populations being compared

```
points['% Point Change in LatinX Pop.'] = (points['LatinX Population 2019']/points['Total Population 2019'])-(points['LatinX Popu
```

```
points['% Point Change in Total Pop.'] = (points['Total Population 2019']-points['Total Population 2016'])/points['Total Populati
```

```
points['% Point Puerto Rican Change'] = (points['Puerto Rican 2019']/points['Total Population 2019'])-(points['Puerto Rican 2016'
```

```
points['% Point Mexican Change'] = (points['Mexican 2019']/points['Total Population 2019'])-(points['Mexican 2016']/points['Total
```

```
points['% Point Cuban Change'] = (points['Cuban 2019']/points['Total Population 2019'])-(points['Cuban 2016']/points['Total Popul
```

```
points['% Point Other LatinX Change'] = (points['Other LatinX 2019']/points['Total Population 2019'])-(points['Other LatinX 2016'
```

```
points
```

Keep necessary attributes only

```
points = points[['Tract','% Point Change in LatinX Pop.','% Point Change in Total Pop.','% Point Puerto Rican Change','% Point Me
```

Add arbitrary columns Latitude and Longitude. This is for converting the pandas dataframe to a geopandas dataframe only. Its only use is for the conversion, you may delete after the conversion.

```
points['Latitude'] = -40.266666
points['Longitude'] = 72.3452
```

```
from geopy.geocoders import Nominatim
```

Convert the pandas dataframe to geopandas using GeoDataFrame() function

```
points = gpd.GeoDataFrame(
    points, geometry=gpd.points_from_xy(points.Longitude, points.Latitude))
```

Import shapefiles for the tracts of MA

```
MA_t = gpd.read_file("CENSUS2010_BLK_BG_TRCT_SHP/CENSUS2010TRACTS_POLY.shp")
```

```
MA_t
```

	STATEFP10	COUNTYFP10	TRACTCE10	GEOID10	NAME10	NAMELSAD10	MTFCC10	ALAND10	AWATER10	INTPTLAT10	...	AREA_ACRES	POP'
0	25	021	418003	25021418003	4180.03	Census Tract 4180.03	G5020	1705668.0	2936.0	+42.2350240	...	422.1740	
1	25	021	417701	25021417701	4177.01	Census Tract 4177.01	G5020	1543651.0	12275.0	+42.2523398	...	384.4502	

Convert the Tracts of MA to match the same format as you created in the previous steps. If the tracts do not match character for character, then it will not make merges successfully.

```
MA_t['NAME10'] = MA_t['NAME10'].astype(float).astype(str)
```

Merge the Tract Geopandas dataframe with the Geopandas Dataframe you created with the population data

```
Tractjoin = gpd.pd.merge(MA_t, points,  how='left', left_on=['NAME10'], right_on = ['Tract'])
```

Drop columns if irrelevant to your data analysis. The columns from the first geopandas dataframe will have the suffix '_x' and the second geopandas dataframe with have the suffix '_y' for its columns that have the same name between the two dataframes.

```
Tractjoin = Tractjoin.drop(columns = ["Latitude","Longitude",'geometry_y'])
```

```
Tractjoin = Tractjoin.rename(columns = {'geometry_x':'geometry'})
```

```
Tractjoin.columns
```

```
Index(['STATEFP10', 'COUNTYFP10', 'TRACTCE10', 'GEOID10', 'NAME10',
       'NAMELSAD10', 'MTFCC10', 'ALAND10', 'AWATER10', 'INTPTLAT10',
       'INTPTLON10', 'AREA_SQFT', 'AREA_ACRES', 'POP100_RE', 'HU100_RE',
       'LOGPL94171', 'LOGSF1', 'LOGACS0610', 'LOGSF1C', 'SHAPE_AREA',
       'SHAPE_LEN', 'geometry', 'Tract', '% Point Change in LatinX Pop.',
       '% Point Change in Total Pop.', '% Point Puerto Rican Change',
       '% Point Mexican Change', '% Point Cuban Change',
       '% Point Other LatinX Change', 'Total Population 2016',
       'LatinX Population 2016', 'Mexican 2016', 'Puerto Rican 2016',
       'Cuban 2016', 'Other LatinX 2016', 'Total Population 2019',
       'LatinX Population 2019', 'Mexican 2019', 'Puerto Rican 2019',
       'Cuban 2019', 'Other LatinX 2019'],
      dtype='object')
```

Drop all NA values.

```
Tractjoin.dropna()
```

Load in the precinct geopandas dataframes.

```
MA_p = gpd.read_file("wardsprecincts_poly/WARDSPRECINCTS_POLY.shp")
MA_p['WARD'] = MA_p['WARD'].astype(str)
```

Load in your csv files containing the election data for each precinct.

```
pres2016 = pd.read_csv('2016PresidentPrecinct.csv')
pres2020 = pd.read_csv('2020PresidentPrecint.csv')
```

Make selections for your columns and rename columns if necessary.

```
pres2016 = pres2016.rename(columns = {'Clinton/ Kaine':'Democratic 2016','Trump/ Pence':'Republican 2016'})
pres2016 = pres2016[['City/Town','Pct','Ward','Democratic 2016','Republican 2016','Total Votes Cast']]
pres2020 = pres2020.rename(columns = {'Democratic': 'Democratic 2020','Republican':'Republican 2020','Precinct':'Pct','Total Vote
pres2020 = pres2020[['City/Town','Pct','Ward','Democratic 2020','Republican 2020','Total Votes Cast 2020']]
```

Drop all NA values.

```
pres2020 = pres2020.dropna()
pres2016 = pres2016.dropna()
```

Merge the two dataframes that contain the election results from each election.

```
mergedP = pd.merge(pres2016,pres2020, how = 'inner', left_on = ['City/Town','Pct','Ward'],right_on = ['City/Town','Pct','Ward'])
```

```
type(mergedP)
```

```
pandas.core.frame.DataFrame
```

Create Lat and Lon columns for the geopandas merge function only.

```
mergedP['Latitude'] = -40.266666
mergedP['Longitude'] = 72.3452
```

Convert the merged dataframe to a geopandas dataframe.

```
mergedPG = gpd.GeoDataFrame(
   mergedP, geometry=gpd.points_from_xy(mergedP.Longitude, mergedP.Latitude))
```

```
mergedPG = mergedPG.drop(columns = ["Latitude","Longitude"])
```

```
type(mergedPG)
```

```
geopandas.geodataframe.GeoDataFrame
```

Review both of the election results and your shapefiles to help make exact match for every City, Precinct, and Ward name. If any values that should be equal are not matching exactly, the merge will fail.

```
mergedPG['City/Town']=mergedPG['City/Town'].str.upper()
```

Create any additional columns you may find useful for your analysis.

```
mergedPG['% Point Change in Democratic Votes'] = (mergedPG['Democratic 2020']/mergedPG['Total Votes Cast 2020'])-(mergedPG['Democ
```

```
mergedPG['% Change in Total Votes'] = (mergedPG['Total Votes Cast 2020']-mergedPG['Total Votes Cast'])/mergedPG['Total Votes Cast
```

```
mergedPG['% Point Change in Republican Votes'] = (mergedPG['Republican 2020']/mergedPG['Total Votes Cast 2020'])-(mergedPG['Repub
```

```
finalP = mergedPG.copy()
```

Drop all NA values.

```
finalP.dropna()
```

Load in Precinct shapefile.

```
MA_p = gpd.read_file("wardsprecincts_poly/WARDSPRECINCTS_POLY.shp")
```

Make necessary conversions as discussed previously.

```
MA_p['WARD'] = MA_p['WARD'].astype(str)
```

```
MA_p['WARD'] = MA_p['WARD'].str.replace('None','-')
```

Merge the shapefile geopandas dataframe with the Precinct Geopandas dataframe you created.

```
joinP = gpd.pd.merge(MA_p, finalP,  how='left', left_on=['PRECINCT','WARD','TOWN'], right_on = ['Pct','Ward','City/Town'])
```

```
joinP
```

	WP_NAME	WARD	PRECINCT	DISTRICT	POP_2010	TOWN	TOWN_ID	AREA_SQMI	AREA_ACRES	YEAR	...	Democratic 2016	Republican 2016	Total Votes Cast	Der
0	Braintree Town Precinct 5B	-	5B	5B	2883	BRAINTREE	40	1.217000	778.800000	2012	...	NaN	NaN	NaN	
1	Braintree Town Precinct 6A	-	6A	6A	3070	BRAINTREE	40	1.994000	1275.960000	2012	...	NaN	NaN	NaN	

```
joinP=joinP.drop(columns = 'geometry_y')
joinP=joinP.rename(columns = {'geometry_x':'geometry'})
```

```
joinP = joinP.dropna()
```

Use a spatial merge to combine the two geopandas dataframes.

```
FinalData = gpd.sjoin(Tractjoin, joinP, how="left", op="intersects")
```

```
FinalData.columns
```

```
Index(['STATEFP10', 'COUNTYFP10', 'TRACTCE10', 'GEOID10', 'NAME10',
       'NAMELSAD10', 'MTFCC10', 'ALAND10', 'AWATER10', 'INTPTLAT10',
       'INTPTLON10', 'AREA_SQFT', 'AREA_ACRES_left', 'POP100_RE', 'HU100_RE',
       'LOGPL94171', 'LOGSF1', 'LOGACS0610', 'LOGSF1C', 'SHAPE_AREA_left',
       'SHAPE_LEN_left', 'geometry', 'Tract', '% Point Change in LatinX Pop.',
       '% Point Change in Total Pop.', '% Point Puerto Rican Change',
       '% Point Mexican Change', '% Point Cuban Change',
       '% Point Other LatinX Change', 'Total Population 2016',
       'LatinX Population 2016', 'Mexican 2016', 'Puerto Rican 2016',
       'Cuban 2016', 'Other LatinX 2016', 'Total Population 2019',
       'LatinX Population 2019', 'Mexican 2019', 'Puerto Rican 2019',
       'Cuban 2019', 'Other LatinX 2019', 'index_right', 'WP_NAME', 'WARD',
       'PRECINCT', 'DISTRICT', 'POP_2010', 'TOWN', 'TOWN_ID', 'AREA_SQMI',
       'AREA_ACRES_right', 'YEAR', 'SHAPE_AREA_right', 'SHAPE_LEN_right',
       'City/Town', 'Pct', 'Ward', 'Democratic 2016', 'Republican 2016',
       'Total Votes Cast', 'Democratic 2020', 'Republican 2020',
       'Total Votes Cast 2020', '% Point Change in Democratic Votes',
       '% Change in Total Votes', '% Point Change in Republican Votes'],
      dtype='object')
```

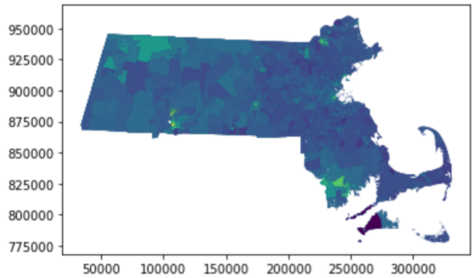
The final merge is complete and further analyses may be performed on the data.

Examples:

Geospatial Analysis

```
FinalData.plot(column = '% Point Change in Republican Votes')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd071cc5580>
```




```
Lneg = points.loc[(points['% Point Change in LatinX Pop.'] < 0)]
Lpos = points.loc[(points['% Point Change in LatinX Pop.'] > 0)]
```

```
Lneg = Lneg[['Tract']]
Lpos = Lpos[['Tract']]

Lneg['Tract'] = Lneg['Tract'].astype(float).astype(str)
Lpos['Tract'] = Lpos['Tract'].astype(float).astype(str)
```

```
Lpos['Latitude'] = -40.266666
Lpos['Longitude'] = 72.3452

Lneg['Latitude'] = -40.266666
Lneg['Longitude'] = 72.3452
```

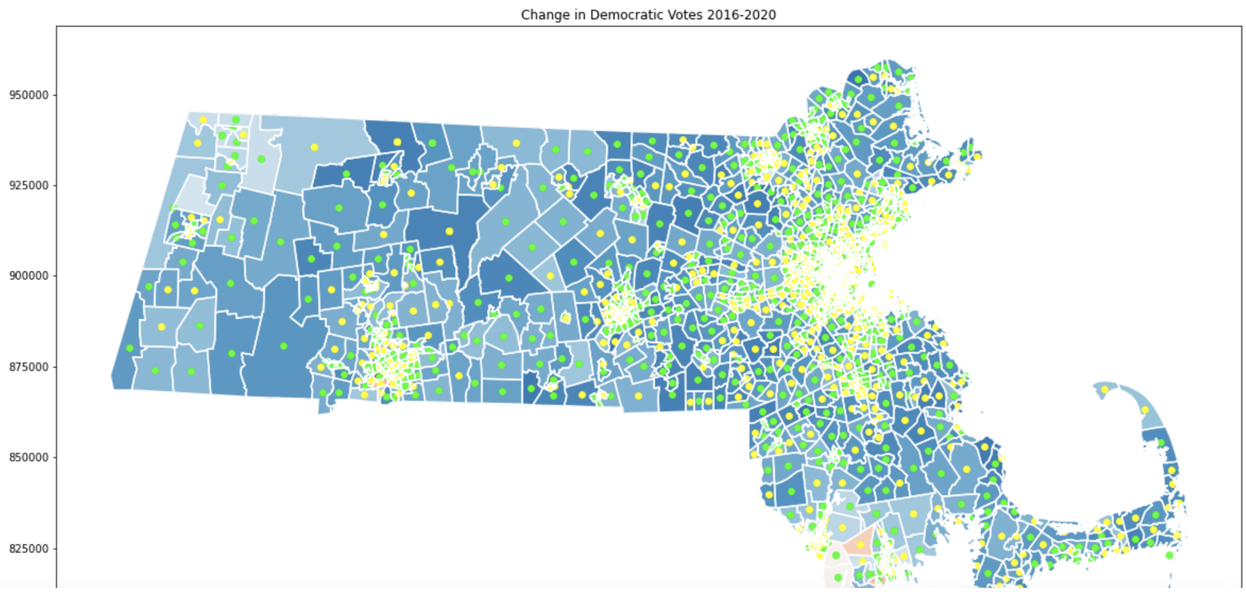
```
Lpos = gpd.GeoDataFrame(
    Lpos, geometry=gpd.points_from_xy(Lpos.Longitude, Lpos.Latitude))

Lneg = gpd.GeoDataFrame(
    Lneg, geometry=gpd.points_from_xy(Lneg.Longitude, Lneg.Latitude))
```

```
LTpos = gpd.pd.merge(MA_t, Lpos,  how='left', left_on=['NAME10'], right_on = ['Tract'])

LTneg = gpd.pd.merge(MA_t, Lneg,  how='left', left_on=['NAME10'], right_on = ['Tract'])
```

```
LTpos = LTpos.dropna()

LTneg = LTneg.dropna()
```

```
import matplotlib.pyplot as plt
fig, ax = plt.subplots(figsize = (20,20))
ax.set_aspect('equal')
ax = MA_t.boundary.plot(ax = ax, color = 'white')
plt.title('Change in Democratic Votes 2016-2020')
FinalData.plot(column='% Point Change in Democratic Votes',cmap = 'RdBu',ax=ax,legend=True,legend_kwds={'label': "% Change in Dem
LTpos.centroid.plot(ax =ax, color = 'lime', marker = 'o', markersize =35, label = 'Increase in LatinX Population')
LTneg.centroid.plot(ax = ax, color = 'yellow', marker = 'o', markersize = 35, label = 'Decrease in LatinX Population')
plt.legend(loc = 'lower left',prop={'size': 20})
```

```
<matplotlib.legend.Legend at 0x7fd06713ed60>
```



Regression Cleaning:

```
#cleaning for presidential election

pres2020 = pres2020.replace(",", "", regex = True)

pres2016 = pres2016.replace(",", "", regex = True)

pres2020["Democratic 2020"] = pd.to_numeric(pres2020["Democratic 2020"])

pres2020["Republican 2020"] = pd.to_numeric(pres2020["Republican 2020"])

pres2020["Total Votes Cast 2020"] = pd.to_numeric(pres2020["Total Votes Cast 2020"])

pres2020["Mexican 2019"] = pd.to_numeric(pres2020["Mexican 2019"])

pres2020["Puerto Rican 2019"] = pd.to_numeric(pres2020["Puerto Rican 2019"])

pres2020["Cuban 2019"] = pd.to_numeric(pres2020["Cuban 2019"])

pres2020["Other LatinX 2019"] = pd.to_numeric(pres2020["Other LatinX 2019"])

pres2020["Total Population 2019"] = pd.to_numeric(pres2020["Total Population 2019"])

pres2016["Democratic 2016"] = pd.to_numeric(pres2016["Democratic 2016"])

pres2016["Republican 2016"] = pd.to_numeric(pres2016["Republican 2016"])

pres2016["Total Votes Cast"] = pd.to_numeric(pres2016["Total Votes Cast"])

pres2016["Mexican 2016"] = pd.to_numeric(pres2016["Mexican 2016"])

pres2016["Puerto Rican 2016"] = pd.to_numeric(pres2016["Puerto Rican 2016"])

pres2016["Cuban 2016"] = pd.to_numeric(pres2016["Cuban 2016"])

pres2016["Other LatinX 2016"] = pd.to_numeric(pres2016["Other LatinX 2016"])

pres2016["Total Population 2016"] = pd.to_numeric(pres2016["Total Population 2016"])
```

Calculating Percentage Point Change:

```
#analyzing changes between the most recent elections for president

changeDemPres = (pres2020["Democratic 2020"] / pres2020["Total Votes Cast 2020"]) - (pres2016["Democratic 2016"] / pres2016["Total Votes Cast"])

changeRepPres = (pres2020["Republican 2020"] / pres2020["Total Votes Cast 2020"]) - (pres2016["Republican 2016"] / pres2016["Total Votes Cast"])

changeMexPres = (pres2020["Mexican 2019"] / pres2020["Total Population 2019"]) - (pres2016["Mexican 2016"] / pres2016["Total Population 2016"])

changePRPres = (pres2020["Puerto Rican 2019"] / pres2020["Total Population 2019"]) - (pres2016["Puerto Rican 2016"] / pres2016["Total Population 2016"])

changeCubanPres = (pres2020["Cuban 2019"] / pres2020["Total Population 2019"]) - (pres2016["Cuban 2016"] / pres2016["Total Population 2016"])

changeOtherLatinXPres = (pres2020["Other LatinX 2019"] / pres2020["Total Population 2019"]) - (pres2016["Other LatinX 2016"] / pres2016["Total Population 2016"])

changeTotalLatinXPres = ((pres2020["Mexican 2019"] + pres2020["Puerto Rican 2019"] + pres2020["Cuban 2019"] + pres2020["Other LatinX 2019"])
 / pres2020["Total Population 2019"]) - ((pres2016["Mexican 2016"] + pres2016["Puerto Rican 2016"] + pres2016["Cuban 2016"] +pres2016["Other LatinX 2016"])
 / pres2016["Total Population 2016"])
```

Merging and creating DataFrame object for each election:

```
#merge presidential election changes with demographic changes during those years

merged_presidential = pd.DataFrame({"change in democratic support" : changeDemPres, "change in republican support" : changeRepPres, "change in mexican population" : changeMexP
                                    "change in puerto rican population" : changePRPres, "change in cuban population" : changeCubanPres, "change in other LatinX population" : c
                                    "change in total LatinX population" : changeTotalLatinXPres})

merged_presidential = merged_presidential.replace([np.inf, -np.inf], np.nan)    #replace inf values by NaN, occurs when starting value is zero

merged_presidential = merged_presidential.fillna(0)     #repalce any NaN with 0
```

Create Scatter Plot, Plot Line of Best Fit, and Include R^2 :

```
#plot presidential election data for mexican population

plt.scatter(merged_presidential["change in mexican population"], merged_presidential["change in democratic support"], color = "blue", label = "democratic support", s= 0.7)

plt.scatter(merged_presidential["change in mexican population"], merged_presidential["change in republican support"], color = "red", label = "republican support", s=0.7)

mexPop = merged_presidential["change in mexican population"].values.reshape(-1, 1)

demSupport = merged_presidential["change in democratic support"].values.reshape(-1, 1)

linregDem = LinearRegression()

linregDem.fit(mexPop, demSupport)

Y_predMexDem = linregDem.predict(mexPop)

mexDemR2 = r2_score(demSupport, Y_predMexDem)

plt.figtext(.7, .73, ("democratic R² =" + "{:.8f}".format(mexDemR2)))

plt.plot(mexPop, Y_predMexDem, color='cyan', label = "democratic support trend")

repSupport = merged_presidential["change in republican support"].values.reshape(-1, 1)

linregRep = LinearRegression()

linregRep.fit(mexPop, repSupport)

Y_predMexRep = linregRep.predict(mexPop)

mexRepR2 = r2_score(repSupport, Y_predMexRep)

plt.figtext(.7, .71, ("republican R²=" + "{:.8f}".format(mexRepR2)))

plt.plot(mexPop, Y_predMexRep, color='lightsalmon', label = "republican support trend")

plt.legend(loc="upper right")

plt.xlabel("% point change in mexican population")

plt.ylabel("% point change in presidential political support")

plt.title("change in mexican population vs change in presidential support (MA 2016 - 2020)")

plt.show()
```

Statistical Analysis:

Read in merged CSV files

In [1]:
```
import pandas as pd
import numpy as np

pres_data = pd.read_csv('MergedMassData.csv')

gov_data = pd.read_csv('MergedMassDataGov.csv')
```

In [2]:
```
pres_data.columns
```

Out[2]:
```
Index(['STATEFP10', 'COUNTYFP10', 'TRACTCE10', 'GEOID10', 'NAME10',
       'NAMELSAD10', 'MTFCC10', 'ALAND10', 'AWATER10', 'INTPTLAT10',
       'INTPTLON10', 'AREA_SQFT', 'AREA_ACRES_left', 'POP100_RE', 'HU100_RE',
       'LOGPL94171', 'LOGSF1', 'LOGACS0610', 'LOGSF1C', 'SHAPE_AREA_left',
       'SHAPE_LEN_left', 'geometry', 'Tract', '% Point Change in LatinX Pop.',
       '% Point Change in Total Pop.', '% Point Puerto Rican Change',
       '% Point Mexican Change', '% Point Cuban Change',
       '% Point Other LatinX Change', 'Total Population 2016',
       'LatinX Population 2016', 'Mexican 2016', 'Puerto Rican 2016',
       'Cuban 2016', 'Other LatinX 2016', 'Total Population 2019',
       'LatinX Population 2019', 'Mexican 2019', 'Puerto Rican 2019',
       'Cuban 2019', 'Other LatinX 2019', 'index_right', 'WP_NAME', 'WARD',
       'PRECINCT', 'DISTRICT', 'POP_2010', 'TOWN', 'TOWN_ID', 'AREA_SQMI',
       'AREA_ACRES_right', 'YEAR', 'SHAPE_AREA_right', 'SHAPE_LEN_right',
       'City/Town', 'Pct', 'Ward', 'Democratic 2016', 'Republican 2016',
       'Total Votes Cast', 'Democratic 2020', 'Republican 2020',
       'Total Votes Cast 2020', '% Point Change in Democratic Votes',
       '% Change in Total Votes', '% Point Change in Republican Votes'],
      dtype='object')
```

```
pres_city = pres_data[['Tract', 'Total Population 2016', 'LatinX Population 2016',
        'Mexican 2016', 'Puerto Rican 2016', 'Cuban 2016', 'Other LatinX 2016',
        'Total Population 2019', 'LatinX Population 2019', 'Mexican 2019',
        'Puerto Rican 2019', 'Cuban 2019', 'Other LatinX 2019', 'City/Town',
        'Democratic 2016', 'Republican 2016', 'Total Votes Cast',
        'Democratic 2020', 'Republican 2020', 'Total Votes Cast 2020']]
```

```
#convert Tract column to type str

pres_city['Tract'] = pres_city['Tract'].astype(str)
```

```
<ipython-input-4-34c4ffa88167>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.
  pres_city['Tract'] = pres_city['Tract'].astype(str)
```

```
#Drop Tract Duplicates
pres_city = pres_city.drop_duplicates(['Tract'])

#Sum up rows by Town
pres_city = pres_city.groupby('City/Town').sum()
```

Add columns with percent change

```
pres_city['% LatinX 2016'] = pres_city['LatinX Population 2016']/pres_city['Total Population 2016']

pres_city['% LatinX 2019'] = pres_city['LatinX Population 2019']/pres_city['Total Population 2019']


pres_city = pres_city.rename(columns = {'Total Votes Cast':'Total Votes Cast 2016'})


pres_city['% Democratic 2016'] = pres_city['Democratic 2016']/pres_city['Total Votes Cast 2016']

pres_city['% Democratic 2020'] = pres_city['Democratic 2020']/pres_city['Total Votes Cast 2020']


pres_city['% Republican 2016'] = pres_city['Republican 2016']/pres_city['Total Votes Cast 2016']

pres_city['% Republican 2020'] = pres_city['Republican 2020']/pres_city['Total Votes Cast 2020']
```

Add columns with percentage point change

```
pres_city['Democratic Percentage point Change'] = (pres_city['% Democratic 2020'] - pres_city['% Democratic 2016'])*100

pres_city['Republican Percentage point Change'] = (pres_city['% Republican 2020'] - pres_city['% Republican 2016'])*100

pres_city['Change LatinX'] = pres_city['% LatinX 2019'] - pres_city['% LatinX 2016']
```

```
pres_city.head()
```

City/Town	Total Population 2016	LatinX Population 2016	Mexican 2016	Puerto Rican 2016	Cuban 2016	Other LatinX 2016	Total Population 2019	LatinX Population 2019	Mexican 2019	Puerto Rican 2019	...	Total Votes Cast 2020	% LatinX 2016	% LatinX 2019
ABINGTON	16586	361	63	157	0	141	17908	342	98	60	...	5553.0	0.021765	0.019098
ACTON	18514	292	27	47	19	199	19437	709	19	102	...	9784.0	0.015772	0.036477
ACUSHNET	7589	245	13	140	51	41	7718	321	11	194	...	3961.0	0.032284	0.041591
ADAMS	7258	189	16	112	0	61	6748	350	68	218	...	2809.0	0.026040	0.051867
AGAWAM	48369	5396	115	4611	111	559	48276	5607	227	4945	...	18782.0	0.111559	0.116145

5 rows × 27 columns

```
#Find Towns where the LatinX pop was greater than 30% in 2016
towns_LatinX_2016 = pres_city.loc[(pres_city['% LatinX 2016'] >= .30)]
```

```
towns_LatinX_2016.shape
```

```
(6, 27)
```

See which of those towns had majority Republican support

```
towns_LatinX_2016 = towns_LatinX_2016.loc[(towns_LatinX_2016['% Republican 2016'] > towns_LatinX_2016['% Democratic 2016'])]
```

```
towns_LatinX_2016.shape
```

```
(0, 27)
```

```
towns_LatinX_2019 = pres_city.loc[(pres_city['% LatinX 2019'] >= .30)]
```

```
towns_LatinX_2019.head()
```

```
towns_LatinX_2019.shape
```

```
(6, 27)
```

```
#Find Towns where the LatinX pop was greater than 10 in 2016%
towns_10_2016 = pres_city.loc[(pres_city['% LatinX 2016'] >= .10)]
```

```
towns_10_2016.shape
```

```
(39, 27)
```

```
towns_10_2016 = towns_10_2016.loc[(towns_10_2016['% Republican 2016'] > towns_10_2016['% Democratic 2016'])]
```

```
towns_10_2016.shape
```

```
(6, 27)
```

```
towns_10_2016
```

```
#Find Towns where the LatinX pop was greater than 10% in 2019
towns_10_2019 = pres_city.loc[(pres_city['% LatinX 2019'] >= .10)]
```

```
towns_10_2019.shape
```

```
(36, 27)
```

```
towns_10_2019 = towns_10_2019.loc[(towns_10_2019['% Republican 2020'] > towns_10_2019['% Democratic 2020'])]
```

```
towns_10_2019
```

City/Town	Total Population 2016	LatinX Population 2016	Mexican 2016	Puerto Rican 2016	Cuban 2016	Other LatinX 2016	Total Population 2019	LatinX Population 2019	Mexican 2019	Puerto Rican 2019	...	Total Votes Cast 2020	% LatinX 2016	% LatinX 2019	% Democratic 2016
AGAWAM	48369	5396	115	4611	111	559	48276	5607	227	4945	...	18782.0	0.111559	0.116145	0.423589

1 rows × 27 columns

```
#Find Towns 2016 where the LatinX pop is greater than 15%
towns_15_2016 = pres_city.loc[(pres_city['% LatinX 2016'] >= .15)]
```

```
towns_15_2016.shape
```

```
(23, 27)
```

```
towns_15_2016 = towns_15_2016.loc[(towns_15_2016['% Republican 2016'] > towns_15_2016['% Democratic 2016'])]
```


```
towns_15_2016 = towns_15_2016.loc[(towns_15_2016['% Republican 2016'] > towns_15_2016['% Democratic 2016'])]
```


```
towns_15_2016
```

City/Town	Total Population 2016	LatinX Population 2016	Mexican 2016	Puerto Rican 2016	Cuban 2016	Other LatinX 2016	Total Population 2019	LatinX Population 2019	Mexican 2019	Puerto Rican 2019	...	Total Votes Cast 2020	% LatinX 2016	% LatinX 2019	% Democratic 2016	% Democratic 2020	% Republican 2016
WILBRAHAM	24087	4880	345	4111	36	388	23480	4486	300	3667	...	12008.0	0.202599	0.191056	0.442097	0.527898	0.463689

1 rows × 27 columns

```
towns_15_2019 = pres_city.loc[(pres_city['% LatinX 2019'] >= .15)]
```

```
towns_15_2019.shape
```

```
(25, 27)
```

```
towns_15_2019 = towns_15_2019.loc[(towns_15_2019['% Republican 2020'] > towns_15_2019['% Democratic 2020'])]
```

```
towns_15_2019.shape
```

```
(0, 27)
```

Top 10 Towns with the highest LatinX Percentage

```
Top 10 Towns with the highest LatinX Percentage

largest_LatinX = pres_city.nlargest(10, ['% LatinX 2019', 'Republican Percentage point Change'])
```

```
largest_LatinX
```

City/Town	Total Population 2016	LatinX Population 2016	Mexican 2016	Puerto Rican 2016	Cuban 2016	Other LatinX 2016	Total Population 2019	LatinX Population 2019	Mexican 2019	Puerto Rican 2019	...	Total Votes Cast 2020	% LatinX 2016	% LatinX 2019	% Democratic 2016	% Democratic 2020
LAWRENCE	67102	48596	297	13146	447	34706	67842	50706	440	12501	...	15104.0	0.724211	0.747413	0.838300	0.734044
CHELSEA	52441	27712	787	4599	217	22109	54565	31645	1299	4502	...	7131.0	0.528441	0.579951	0.734743	0.766092
HOLYOKE	45058	16675	111	15439	41	1084	45343	18404	217	16586	...	13320.0	0.370079	0.405884	0.673746	0.657583
SPRINGFIELD	167915	62062	2538	51688	273	7563	167224	67004	2087	56767	...	33729.0	0.369604	0.400684	0.728685	0.697975
METHUEN	62471	21879	275	7071	173	14360	64852	25768	255	6979	...	26881.0	0.350227	0.397335	0.535401	0.576615
LYNN	102991	34226	1600	5379	155	27092	103603	38091	1839	3930	...	28753.0	0.332320	0.367663	0.686249	0.701040
EVERETT	39641	9394	202	1166	53	7973	40852	11215	224	1620	...	9545.0	0.236977	0.274528	0.657777	0.692300
REVERE	46091	11178	641	1279	83	9175	45426	12432	542	1862	...	7643.0	0.242520	0.273676	0.568050	0.599764
FITCHBURG	44974	10155	853	6710	34	2558	46329	11950	1060	8446	...	15794.0	0.225797	0.257938	0.561271	0.601051
SHERBORN	24450	6091	576	2225	11	3279	25212	5860	349	2223	...	14955.0	0.249121	0.232429	0.648549	0.757940

10 rows × 27 columns

```
insight = pres_city.nlargest(5, ['% LatinX 2019', 'Republican Percentage point Change'])

insight = insight[['% LatinX 2019', 'Democratic Percentage point Change', 'Republican Percentage point Change']]
```

```
insight
```

City/Town	% LatinX 2019	Democratic Percentage point Change	Republican Percentage point Change
LAWRENCE	0.747413	-10.425567	12.561895
CHELSEA	0.579951	3.134887	4.054185
HOLYOKE	0.405884	-1.616373	6.618964
SPRINGFIELD	0.400684	-3.071012	4.796808
METHUEN	0.397335	4.121485	0.773999

Percentage of cities that saw an increase in the LatinX pop from 2016-2019

```
percent of cities that saw an increase in the LatinX pop from 2016-2019

increase = pres_city[(pres_city['Change LatinX']) > 0].count()[0] / pres_city.shape[0]

decrease = 1 - increase

print("Increase =", increase)
print("Decrease =", decrease)
```

```
Increase = 0.73828125
Decrease = 0.26171875
```

```
#avg LatinX Percentage change for all Towns in MA
pres_city['Change LatinX'].mean()
```

```
0.0075259252143843975
```

```
#avg Percentage change in Republican Support for all Towns in MA
pres_city['Republican Percentage point Change'].mean()
```

```
-0.9359053731964311
```

Percent of cities that saw an increase in the Republican Support from 2016-2019

```
increase = pres_city[(pres_city['Republican Percentage point Change']) > 0].count()[0] / pres_city.shape[0]

decrease = 1 - increase

print("Increase =", increase)
print("Decrease =", decrease)
```

```
Increase = 0.2421875
Decrease = 0.7578125
```