

For my final project, I would like to do an exploratory analysis project! I am taking a Campaigns and Elections course this semester, and we had speculated about the prevalence of split-ticket voting in the 2020 election year. Split ticket voting is when a voter votes for candidates from different political parties for different offices, on the same ballot. Unfortunately, those vote returns from 2020 will not be available for a while, but the 2016 election returns are available! It is still interesting to look at the 2016 election, because state legislatures can have a different partisan leaning from how their state typically votes on the presidential level. For example, the New Hampshire State Legislature was controlled by Republicans 2016.¹ The Democratic Party did not have clear control, which is interesting since New Hampshire has voted for the Democratic presidential nominee in every election since the 2000.²

For my data, I will use five datasets made available by the MIT Election Lab.³ The datasets are the precinct-level vote returns for 2016 for elections at the local, state, and federal level. The federal level elections include the U.S. House, U.S. Senate, and President. I have already loaded these datasets into a Jupyter Lab Notebook and poked around the data. Precinct names are consistent across each of the five datasets, vote returns are listed by party by precinct, and the office name is included in the local dataset.

For my guiding questions, I have three.

1. How prominent was split ticket voting in 2016?
 - a. This question is not particularly quantifiable, but it could be an interesting concluding paragraph. It may also be interesting to compare states based on percentages of precincts where split ticket voting was common, over a certain percentage.
2. If split ticket voting was common, in which level of races did voters most often break with their “majority” party?
 - a. This will be based on the republicans, democratic, and independent candidate vote returns for each level, by precinct. I will then see which level race had the largest change in votes for a party’s candidate. I will likely look at these totals by state, in case there was a particularly polarizing candidate in a lower-level election. I will then look and see if there was a national trend.
3. Was it more common for precincts who voted for majority republicans to split for a democrat candidate? Or was it more common for precinct who voted for majority democrats to split for a republican candidate?
 - a. Unfortunately, individual-level data is not available. But it would be interesting to see how partisan leaning impacts commonality of split ticket voting.

I anticipate my largest difficulty with this project will be aggregating all of the vote returns for each precinct for each election level. I am still working through the best way to do that. I will also struggle to determine a way to deem a precinct “Traditionally Republican” or “Traditionally Democrat” to determine which race was the “split” race, if there was one. While these are

¹ <https://www.ncsl.org/research/about-state-legislatures/partisan-composition.aspx#>

² https://www.270towin.com/states/New_Hampshire

³ <https://electionlab.mit.edu/data>

challenges, I think I will be able to do it! If I get really stuck, I will first take a break and circle back around and then ask for help.

```
[5]: presPrec = pd.read_csv("2016-precinct-president.csv", encoding='ISO-8859-1')
statPrec = pd.read_csv("2016-precinct-state.csv", encoding='ISO-8859-1')
senaPrec = pd.read_csv("2016-precinct-senate.csv", encoding='ISO-8859-1')
locaPrec = pd.read_csv("2016-precinct-local.csv", encoding='ISO-8859-1')
housPrec = pd.read_csv("2016-precinct-house.csv", encoding='ISO-8859-1')
```

```
/opt/miniconda2/envs/DataSci/lib/python3.8/site-packages/IPython/core/interactiveshell.py:3145: DtypeWarni
g: Columns (13,17,19,31,32) have mixed types.Specify dtype option on import or set low_memory=False.
  has_raised = await self.run_ast_nodes(code_ast.body, cell_name,
/opt/miniconda2/envs/DataSci/lib/python3.8/site-packages/IPython/core/interactiveshell.py:3145: DtypeWarni
g: Columns (7,13,17) have mixed types.Specify dtype option on import or set low_memory=False.
  has_raised = await self.run_ast_nodes(code_ast.body, cell_name,
/opt/miniconda2/envs/DataSci/lib/python3.8/site-packages/IPython/core/interactiveshell.py:3145: DtypeWarni
g: Columns (13,19,27,30,32) have mixed types.Specify dtype option on import or set low_memory=False.
  has_raised = await self.run_ast_nodes(code_ast.body, cell_name,
/opt/miniconda2/envs/DataSci/lib/python3.8/site-packages/IPython/core/interactiveshell.py:3145: DtypeWarni
g: Columns (13,17,19) have mixed types.Specify dtype option on import or set low_memory=False.
  has_raised = await self.run_ast_nodes(code_ast.body, cell_name,
/opt/miniconda2/envs/DataSci/lib/python3.8/site-packages/IPython/core/interactiveshell.py:3145: DtypeWarni
g: Columns (29,30) have mixed types.Specify dtype option on import or set low_memory=False.
  has_raised = await self.run_ast_nodes(code_ast.body, cell_name,
```

```
[71]: print("Presidential Keys: " + str(presPrec.keys()))
print("State Keys: " + str(statPrec.keys()))
print("Local Keys: " + str(locaPrec.keys()))
print("House Keys: " + str(housPrec.keys()))
print("Senate Keys: " + str(senaPrec.keys()))
```

```
Presidential Keys: Index(['year', 'stage', 'special', 'state', 'state_postal', 'state_fips',
'state_icpsr', 'county_name', 'county_fips', 'county_ansi',
```