

CH03_Wisconsin_Elections_Python_Supplement

September 3, 2026

1 Chapter 3 — Wisconsin elections (Python supplement)

Condensed notebook for the **Python** portion of *Chapter 3: Wisconsin Election Data*.

Target graphics:

- Matplotlib ward choropleths (**Figures 3.67–3.71**)
- County change choropleth and cartogram (**Figures 3.72–3.73**)
- Plotly geographic bubble maps (**Figures 3.74–3.75**)
- County and ward scatterplots (**Figures 3.76–3.77**)

Data: 2012–2020_Election_Data_with_2020_Wards/ shapefile folder and preprocessed pres_election_data_12_16.geojson in Data For Condensed Notebooks.

Dependencies:

- geopandas
- pandas
- matplotlib
- geoplot
- plotly
- numpy

1.1 Imports and display options

```
[1]: import geopandas as gpd
import pandas as pd
import matplotlib.pyplot as plt
import geoplot as gplt
import plotly.express as px
import plotly.graph_objects as go
import plotly.io as pio
import numpy as np
```

```
[2]: pio.renderers.default = "pdf+jupyterlab+notebook"
```

1.2 Data prep

Preprocess the shapefile and save a smaller GeoJSON for faster loading.

Load the source file (may take several minutes).

```
[3]: gdf = gpd.read_file('../Data For Condensed Notebooks/
↳2012-2020_Election_Data_with_2020_Wards/
↳2012-2020_Election_Data_with_2020_Wards.shp')
gdf
```

```
[3]:
```

	OBJECTID	GEOID	CNTY_FIPS	CNTY_NAME	COUSUBFP	MCD_FIPS	\
0	1	55001002750001	55001	Adams	00275	5500100275	
1	2	55001002750002	55001	Adams	00275	5500100275	
2	3	55001002750003	55001	Adams	00275	5500100275	
3	4	55001002750004	55001	Adams	00275	5500100275	
4	5	55001003000001	55001	Adams	00300	5500100300	
...	...	...	...	...	...	...	
7073	7075	55139557500027	55139	Winnebago	55750	5513955750	
7074	7076	55139557500028	55139	Winnebago	55750	5513955750	
7075	7077	55139598750009	55139	Winnebago	59875	5513959875	
7076	7078	55139605000043	55139	Winnebago	60500	5513960500	
7077	7201	55065081250001	55065	Lafayette	08125	5506508125	

	MCD_NAME	CTV		LABEL	LSAD	...	WSASCT12	WSSTOT12	\
0	Adams	C	Adams - C	0001	25	...	0.0	230.0	
1	Adams	C	Adams - C	0002	25	...	0.0	224.0	
2	Adams	C	Adams - C	0003	25	...	0.0	100.0	
3	Adams	C	Adams - C	0004	25	...	0.0	125.0	
4	Adams	T	Adams - T	0001	43	...	0.0	335.0	
...	...	..	...	...	...	...	...	...	
7073	Neenah	C	Neenah - C	0027	25	...	NaN	NaN	
7074	Neenah	C	Neenah - C	0028	25	...	NaN	NaN	
7075	Omro	C	Omro - C	0009	25	...	NaN	NaN	
7076	Oshkosh	C	Oshkosh - C	0043	25	...	NaN	NaN	
7077	Blanchardville	V	Blanchardville - V	0001	47	...	0.0	0.0	

	WSSDEM12	WSSREP12	WSSREP212	WSSCON12	WSSIND12	WSSSCT12	WSSAME12	\
0	137.0	92.0	0.0	0.0	0.0	1.0	0.0	
1	135.0	89.0	0.0	0.0	0.0	0.0	0.0	
2	60.0	40.0	0.0	0.0	0.0	0.0	0.0	
3	75.0	50.0	0.0	0.0	0.0	0.0	0.0	
4	185.0	150.0	0.0	0.0	0.0	0.0	0.0	
...	...	...	...	...	...	...	...	
7073	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
7074	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
7075	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
7076	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
7077	0.0	0.0	0.0	0.0	0.0	0.0	0.0	

```
geometry
0 POLYGON ((1751356.141 1273629.537, 1751356.133...
1 POLYGON ((1756200.655 1268370.257, 1755173.241...
```

```

2     POLYGON ((1759268.232 1273626.025, 1759271.923...
3     POLYGON ((1754028.062 1270988.986, 1752360.561...
4     POLYGON ((1780684.08 1247327.681, 1780605.34 1...
...
7073  POLYGON ((2093474.457 1339931.004, 2093488.304...
7074  POLYGON ((2094994.454 1343954.9, 2094685.554 1...
7075  POLYGON ((2038923.585 1299696.362, 2038730.382...
7076  POLYGON ((2090087.073 1281654.313, 2089968.582...
7077  POLYGON ((1744675.771 851748.312, 1744686.704 ...

```

[7078 rows x 275 columns]

```
[4]: gdf.set_index('OBJECTID', inplace=True)
```

```
[5]: gdf.geometry.crs
```

```
[5]: <Projected CRS: ESRI:102219>
Name: NAD_1983_Wisconsin_TM_US_Ft
Axis Info [cartesian]:
- [east]: Easting (US survey foot)
- [north]: Northing (US survey foot)
Area of Use:
- undefined
Coordinate Operation:
- name: unnamed
- method: Transverse Mercator
Datum: North American Datum 1983
- Ellipsoid: GRS 1980
- Prime Meridian: Greenwich
```

```
[6]: gdf.to_crs('EPSG:4326', inplace=True)
```

```
[7]: gdf.columns
```

```
[7]: Index(['GEOID', 'CNTY_FIPS', 'CNTY_NAME', 'COUSUBFP', 'MCD_FIPS', 'MCD_NAME',
        'CTV', 'LABEL', 'LSAD', 'NAME',
        ...,
        'WSASCT12', 'WSTOT12', 'WSSDEM12', 'WSSREP12', 'WSSREP212', 'WSSCON12',
        'WSSIND12', 'WSSSCT12', 'WSSAME12', 'geometry'],
        dtype='str', length=274)
```

Select presidential-election columns and compute `prop_dem_12 / prop_dem_16`.

```
[8]: pres_gdf = gdf[['CNTY_FIPS', 'CNTY_NAME',
                    'PREREP12', 'PREDEM12', 'PREREP16', 'PREDEM16',
                    'PERSONS', 'geometry']].copy()

pres_gdf
```

```
[8]:
```

	CNTY_FIPS	CNTY_NAME	PREREP12	PREDEM12	PREREP16	PREDEM16	\
OBJECTID							
1	55001	Adams	101.0	168.0	127.0	110.0	
2	55001	Adams	98.0	164.0	124.0	107.0	
3	55001	Adams	44.0	73.0	56.0	48.0	
4	55001	Adams	55.0	92.0	70.0	60.0	
5	55001	Adams	181.0	240.0	253.0	139.0	
...	...	...	...	...	...	...	
7075	55139	Winnebago	NaN	NaN	NaN	NaN	
7076	55139	Winnebago	NaN	NaN	NaN	NaN	
7077	55139	Winnebago	NaN	NaN	NaN	NaN	
7078	55139	Winnebago	NaN	NaN	NaN	NaN	
7201	55065	Lafayette	105.0	230.0	105.0	178.0	

```
PERSONS
```

		geometry
OBJECTID		
1	661.0	POLYGON ((-89.82779 43.96683, -89.82779 43.966...
2	652.0	POLYGON ((-89.80943 43.95237, -89.81333 43.952...
3	288.0	POLYGON ((-89.79773 43.96677, -89.79771 43.965...
4	366.0	POLYGON ((-89.81766 43.95957, -89.82399 43.959...
5	902.0	POLYGON ((-89.7167 43.89443, -89.717 43.89443,...
...	...	...
7075	NaN	POLYGON ((-88.5236 44.13937, -88.52358 44.1380...
7076	NaN	POLYGON ((-88.51753 44.15033, -88.51871 44.150...
7077	NaN	POLYGON ((-88.73377 44.03148, -88.73451 44.031...
7078	NaN	POLYGON ((-88.54043 43.97966, -88.54088 43.979...
7201	648.0	POLYGON ((-89.85594 42.80899, -89.8559 42.8091...

[7078 rows x 8 columns]

```
[9]: pres_gdf['prop_dem_12'] = pres_gdf['PREDEM12'] / (pres_gdf['PREDEM12'] +
pres_gdf['PREREP12'])
pres_gdf['prop_dem_16'] = pres_gdf['PREDEM16'] / (pres_gdf['PREDEM16'] +
pres_gdf['PREREP16'])
```

Some wards have no major-party votes, so proportions are NaN. Use `dropna=False` to see them.

```
[10]: pres_gdf['prop_dem_12'].value_counts(dropna=False)
```

```
[10]: prop_dem_12
NaN      420
0.500000  117
0.666667   28
0.428571   19
0.000000   18
...
0.523944    1
0.521605    1
```

```

0.521127      1
0.435730      1
0.795400      1
Name: count, Length: 5460, dtype: int64

```

```
[11]: pres_gdf['prop_dem_16'].value_counts(dropna=False)
```

```

[11]: prop_dem_16
NaN      429
0.500000   58
0.333333   51
0.000000   31
1.000000   28
...
0.421212    1
0.421405    1
0.313901    1
0.764296    1
0.628975    1
Name: count, Length: 5368, dtype: int64

```

```
[12]: print(pres_gdf[pres_gdf['prop_dem_12'].isna()]['PREDEM12'].sum())
print(pres_gdf[pres_gdf['prop_dem_12'].isna()]['PREREP12'].sum())
print(pres_gdf[pres_gdf['prop_dem_16'].isna()]['PREDEM16'].sum())
print(pres_gdf[pres_gdf['prop_dem_16'].isna()]['PREREP16'].sum())
```

```

0.0
0.0
0.0
0.0

```

```
[13]: print(pres_gdf[pres_gdf['prop_dem_12'].isna()]['PERSONS'].sum())
print(pres_gdf[pres_gdf['prop_dem_16'].isna()]['PERSONS'].sum())
```

```

658.0
720.0

```

```
[14]: pres_gdf.to_file('../Data For Condensed Notebooks/pres_election_data_12_16.
↳geojson')
```

1.3 Graphics — load preprocessed data

```
[15]: pres_gdf = gpd.read_file('../Data For Condensed Notebooks/
↳pres_election_data_12_16.geojson', index_col='OBJECTID')
pres_gdf.set_index('OBJECTID', inplace=True)
```

```

/Users/mckenzielamb/Library/CloudStorage/Dropbox/Academics/Data Vis
Book/.venv/lib/python3.12/site-packages/pyogrio/raw.py:200: RuntimeWarning:

```

```
driver GeoJSON does not support open option INDEX_COL
return ogr_read(
```

1.4 Choropleths by ward (Matplotlib)

Plotly cannot efficiently map all ward geometries; use GeoPandas plot instead. **Figures 3.67–3.71.**

Figure 3.67 — Initial choropleth with built-in RdBu colormap.

```
[16]: pres_gdf.plot(column='prop_dem_12', cmap='RdBu', legend=True)
plt.show()
```

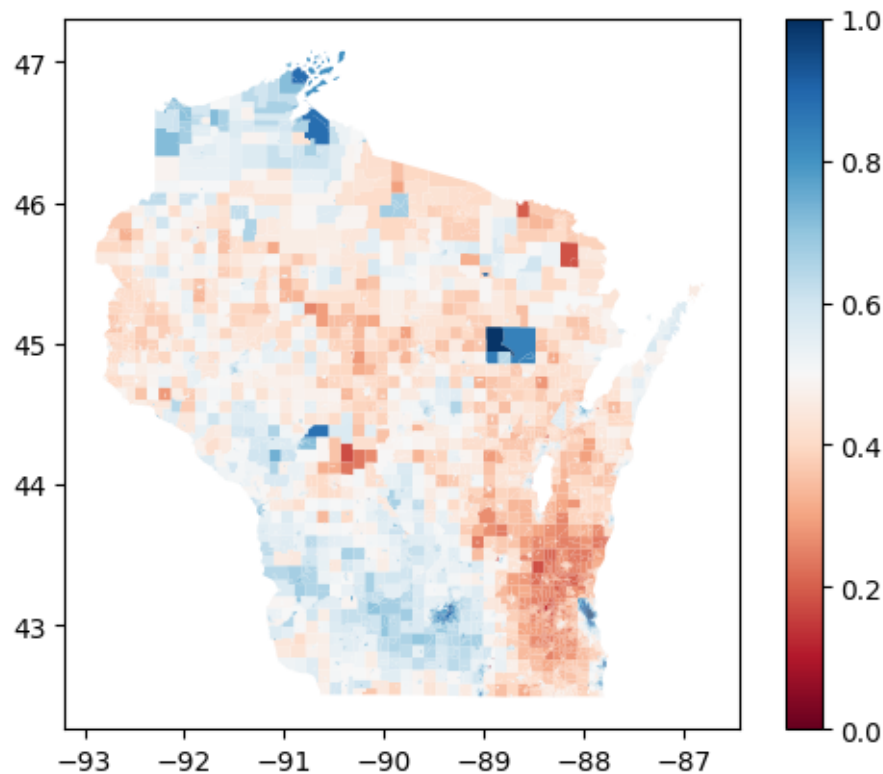


Figure 3.68 — Custom red-to-blue colormap.

```
[17]: from matplotlib.colors import LinearSegmentedColormap

colors = [(1, 0, 0), (0, 0, 1)]
rb_cmap = LinearSegmentedColormap.from_list('RedBlue', colors)

pres_gdf.plot(column='prop_dem_12', cmap=rb_cmap)
plt.show()
```

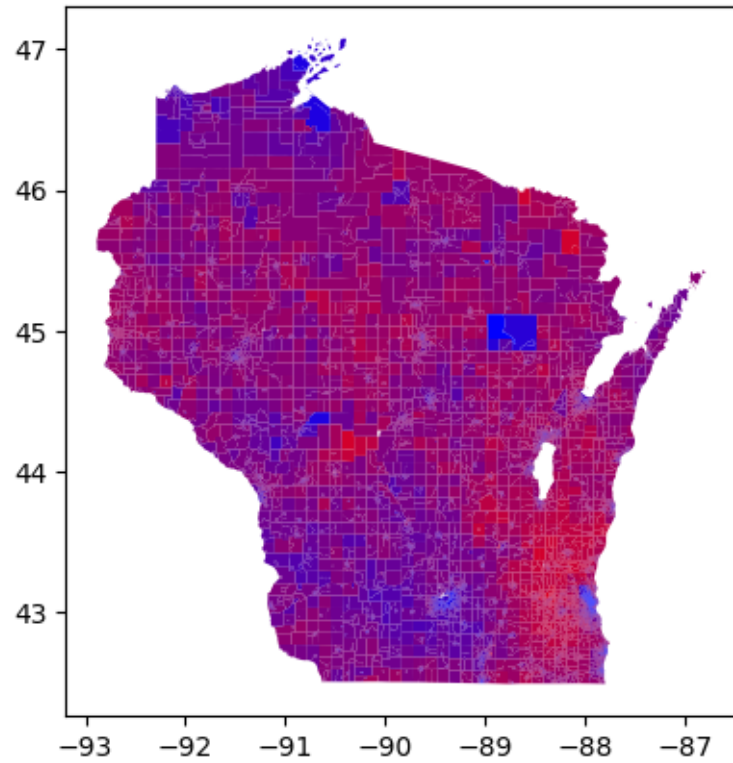


Figure 3.69 — Highlight missing values in yellow.

```
[18]: pres_gdf.plot(column='prop_dem_12', cmap=rb_cmap,
                  vmin=0, vmax=1,
                  missing_kwds={'color': 'yellow',
                              'edgecolor': 'yellow',
                              'label': 'Missing values'},
                  legend=True)
plt.show()
```

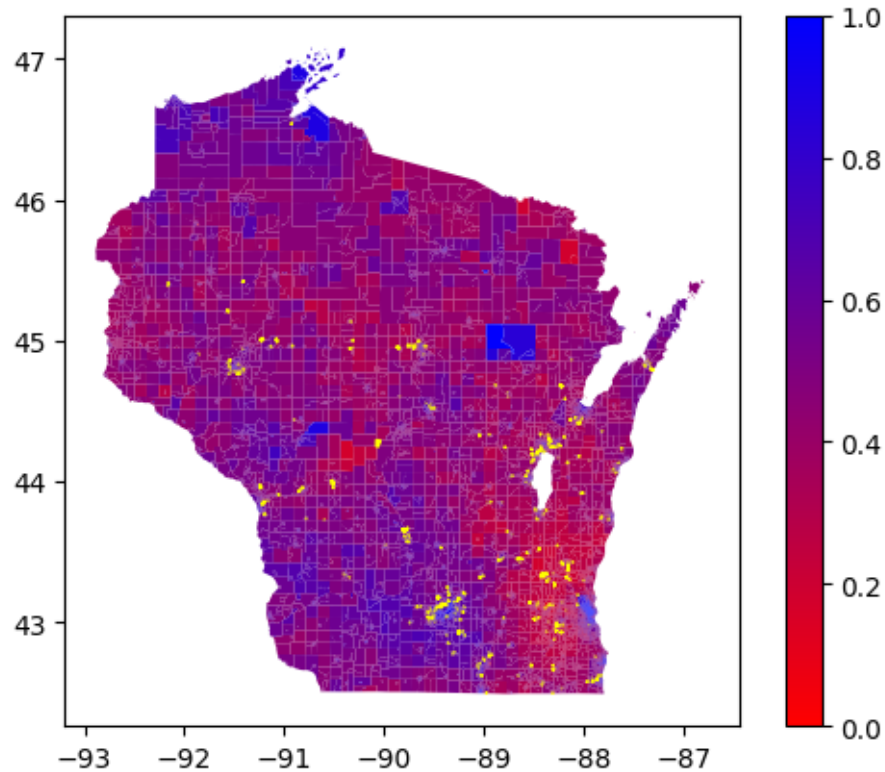


Figure 3.70 — Side-by-side 2012 and 2016 maps (draft).

```
[19]: fig, (ax1, ax2) = plt.subplots(figsize=(10, 4), ncols=2)

map12 = pres_gdf.plot(column='prop_dem_12', cmap=rb_cmap,
                      vmin=0, vmax=1,
                      missing_kwds={'color': 'yellow',
                                   'edgecolor': 'yellow',
                                   'label': 'Missing values'}, ax=ax1)
map16 = pres_gdf.plot(column='prop_dem_16', cmap=rb_cmap,
                      vmin=0, vmax=1,
                      missing_kwds={'color': 'yellow',
                                   'edgecolor': 'yellow',
                                   'label': 'Missing values'}, ax=ax2)
```

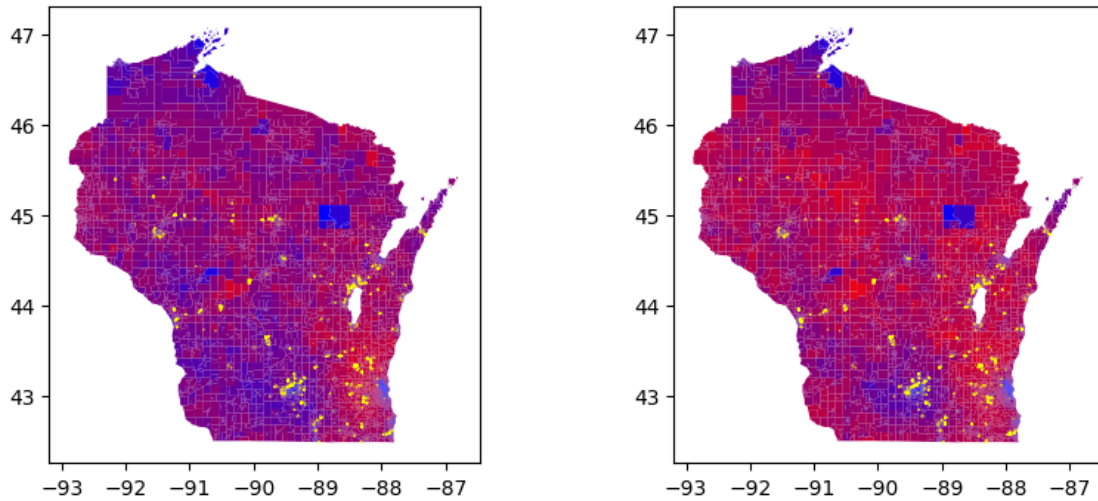


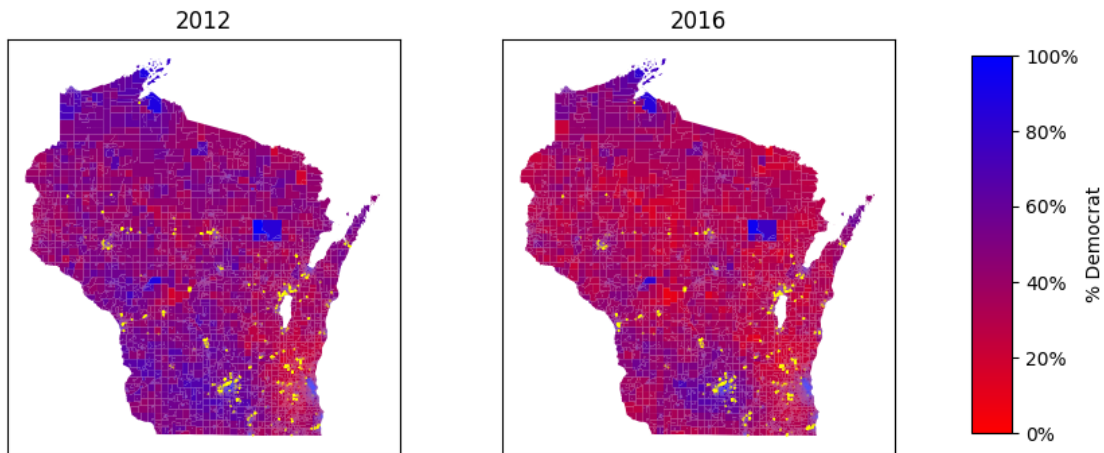
Figure 3.71 — Shared colorbar and titles.

```
[20]: from IPython.display import display
      from matplotlib.ticker import PercentFormatter

      fig.subplots_adjust(right=0.8)
      cbar_ax = fig.add_axes([0.85, 0.15, 0.03, 0.7])
      sm = plt.cm.ScalarMappable(cmap=rb_cmap, norm=plt.Normalize(vmin=0, vmax=1))
      fig.colorbar(sm,
                  cax=cbar_ax,
                  format=PercentFormatter(xmax=1, decimals=0),
                  label='% Democrat')

      ax1.set_xticks([])
      ax1.set_yticks([])
      ax2.set_xticks([])
      ax2.set_yticks([])
      ax1.set_title('2012')
      ax2.set_title('2016')
      fig.suptitle('Comparing Elections', fontsize=16, va='bottom', ha='center')
      display(fig)
```

Comparing Elections



1.5 Aggregate by county

First attempt averages ward proportions (incorrect); then sum votes and recompute.

```
[21]: by_county = pres_gdf.dissolve(by='CNTY_NAME',
                                     aggfunc={'CNTY_FIPS': 'first',
                                               'prop_dem_12': 'mean',
                                               'prop_dem_16': 'mean',
                                               'PERSONS': 'sum'})

by_county
```

```
[21]:
```

	geometry	CNTY_FIPS	\
CNTY_NAME			
Adams	POLYGON ((-89.84382 43.7559, -89.84382 43.7559...	55001	
Ashland	MULTIPOLYGON (((-90.92436 46.24539, -90.92447 ...	55003	
Barron	POLYGON ((-92.00321 45.20891, -92.00429 45.208...	55005	
Bayfield	MULTIPOLYGON (((-91.18592 46.15737, -91.19631 ...	55007	
Brown	MULTIPOLYGON (((-88.18083 44.24177, -88.18231 ...	55009	
...	...	...	
Waukesha	POLYGON ((-88.42602 42.84252, -88.4261 42.8425...	55133	
Waupaca	MULTIPOLYGON (((-89.10593 44.24309, -89.10839 ...	55135	
Waushara	POLYGON ((-89.44999 43.98227, -89.45514 43.982...	55137	
Winnebago	MULTIPOLYGON (((-88.65818 43.89468, -88.66203 ...	55139	
Wood	MULTIPOLYGON (((-90.09335 44.24913, -90.1034 4...	55141	

	prop_dem_12	prop_dem_16	PERSONS
CNTY_NAME			
Adams	0.545645	0.380557	20875.0
Ashland	0.630363	0.528597	16157.0
Barron	0.486828	0.360468	45870.0

Bayfield	0.630420	0.558975	15014.0
Brown	0.476178	0.428208	248004.0
...	...	...	...
Waukesha	0.317801	0.343545	389891.0
Waupaca	0.456589	0.335568	52410.0
Waushara	0.450668	0.329693	24496.0
Winnebago	0.499319	0.429739	166994.0
Wood	0.482886	0.390960	74749.0

[72 rows x 5 columns]

```
[22]: by_county2 = pres_gdf.dissolve(by='CNTY_NAME',
                                     aggfunc={'CNTY_FIPS': 'first',
                                               'CNTY_NAME': 'first',
                                               'PREDEM12': 'sum',
                                               'PREREP12': 'sum',
                                               'PREDEM16': 'sum',
                                               'PREREP16': 'sum',
                                               'PERSONS': 'sum'})

by_county2['prop_dem_12'] = by_county2['PREDEM12'] / (by_county2['PREDEM12'] +
↳by_county2['PREREP12'])
by_county2['prop_dem_16'] = by_county2['PREDEM16'] / (by_county2['PREDEM16'] +
↳by_county2['PREREP16'])
by_county2['prop_dem_change'] = by_county2['prop_dem_16'] -
↳by_county2['prop_dem_12']
```

Figure 3.72 — County choropleth of Democratic vote share change (Plotly).

```
[23]: fig = px.choropleth(by_county2,
                           geojson=by_county2.geometry,
                           locations=by_county2.index,
                           color='prop_dem_change',
                           color_continuous_scale='bluered_r',
                           color_continuous_midpoint=0,
                           projection='mercator',
                           labels={'prop_dem_change': 'Percentage Point Change',
                                   'index': 'County'},
                           title='Change from 2012 to 2016',
                           hover_data={'prop_dem_change': ':.3%'},
                           width=700,
                           height=700)

fig.update_geos(fitbounds='locations', visible=False)
fig.update_traces(marker_line_width=0, selector=dict(type='choropleth'))
fig.update_layout(coloraxis_colorbar=dict(lenmode='pixels', len=300))
fig.update_coloraxes(
```

```

colorbar_tickvals=[-0.20, -0.15, -0.10, -0.05, 0, 0.05, 0.1, 0.15, 0.20],
colorbar_ticktext=['-20pp', '-15pp', '-10pp', '-5pp', '0pp', '5pp', '10pp', '15pp', '20pp']
)
fig.show()

```

Change from 2012 to 2016

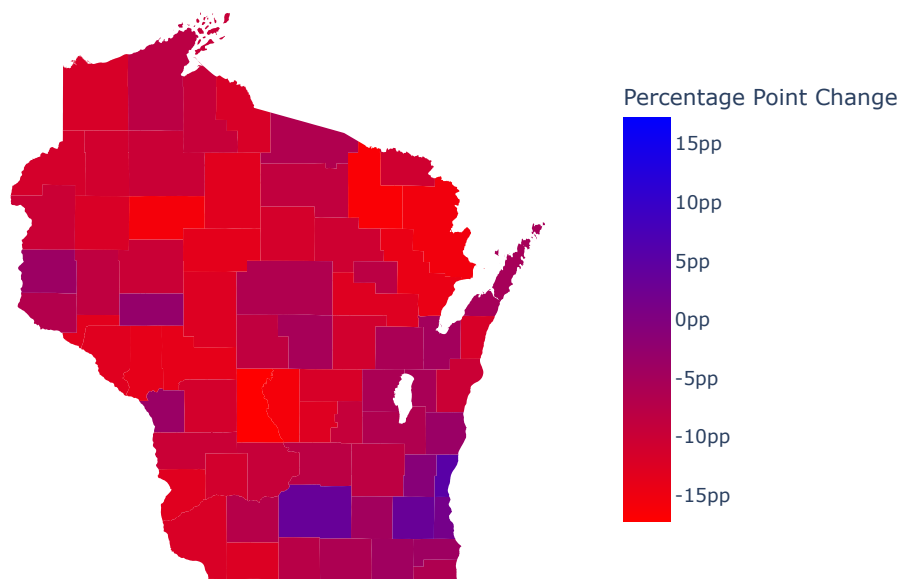


Figure 3.73 — Population-scaled cartogram (geoplot).

```

[24]: from matplotlib.colors import TwoSlopeNorm
import geoplot.crs as gcrs

divnorm = TwoSlopeNorm(vmin=-0.2, vcenter=0, vmax=0.2)

```

```

gplt.cartogram(
    by_county2,
    scale='PERSONS',
    hue='prop_dem_change',
    cmap=rb_cmap,
    norm=divnorm,
    linewidth=0.3,
    edgecolor='white',
    legend=True,
    projection=gcrs.Mercator(),
    legend_var='hue',
    limits=(0.3, 1.3),
    figsize=(10, 10),
    legend_kwargs={'label': 'Percentage Point Change',
                   'orientation': 'vertical',
                   'format': PercentFormatter(xmax=1, decimals=0),
                   'shrink': 0.5}
)
plt.title('Change from 2012 to 2016, Sized by Population')

```

[24]: Text(0.5, 1.0, 'Change from 2012 to 2016, Sized by Population')

Change from 2012 to 2016, Sized by Population



1.6 Geographic bubble chart (Plotly)

Figure 3.74 — Bubbles sized by population, colored by change.

```
[25]: scatter_fig = px.scatter_geo(by_county2,  
    geojson=by_county2.geometry,  
    locations=by_county2.index,  
    color='prop_dem_change',  
    color_continuous_scale='bluered_r',  
    color_continuous_midpoint=0,  
    hover_name='CNTY_FIPS', size='PERSONS',  
    labels={'prop_dem_change': 'Percentage Point Change',  
           'index': 'County'},  
    title='Change from 2012 to 2016',  
    hover_data={'prop_dem_change': ':.3%'},  
    projection='mercator',
```

```

        height=600,
        width=600)

scatter_fig.update_coloraxes(
    colorbar_tickvals=[-0.20, -0.15, -0.10, -0.05, 0, 0.05, 0.1, 0.15, 0.20],
    colorbar_ticktext=['-20pp', '-15pp', '-10pp', '-5pp', '0pp', '5pp', '10pp', '15pp', '20pp']
)
scatter_fig.update_layout(coloraxis_colorbar=dict(lenmode='pixels', len=300))
scatter_fig.update_geos(fitbounds='locations', visible=True)
scatter_fig.show()

```

Change from 2012 to 2016

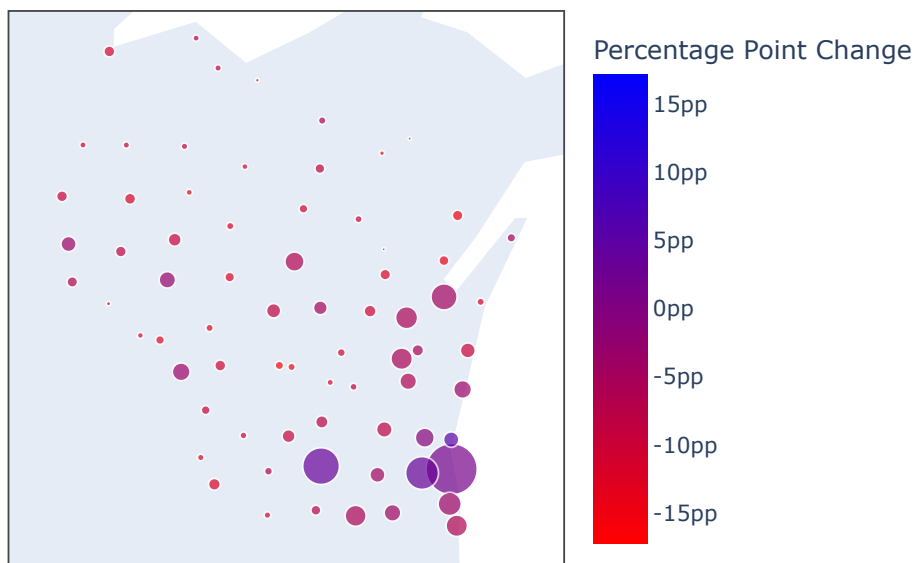


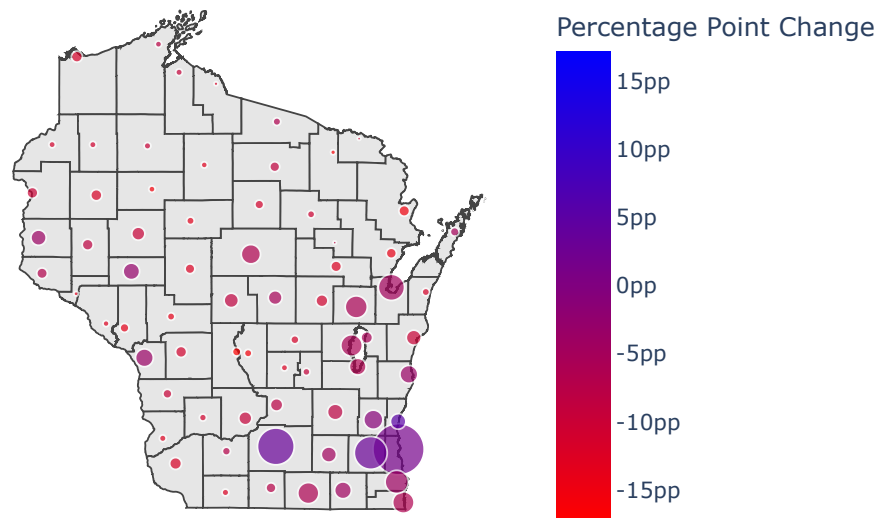
Figure 3.75 — Bubbles over a light county base map.

```
[26]: county_map_fig = px.choropleth(by_county2,
                                     geojson=by_county2.geometry,
                                     locations=by_county2.index,
                                     color=len(by_county2) * ['base_color'],
                                     color_discrete_map={'base_color': 'rgb(0.9, 0.9, 0.9)'},
                                     projection='mercator',
                                     width=700,
                                     height=700)

county_map_fig.update_traces(marker_line_width=1,
                             selector=dict(type='choropleth'),
                             showscale=False)
county_map_fig.update_traces(showlegend=False)

county_map_trace = county_map_fig.data[0]
scatter_fig.add_trace(county_map_trace)
scatter_fig.update_geos(fitbounds='locations', visible=False)
scatter_fig.show()
```

Change from 2012 to 2016



1.7 Scatterplots

Figure 3.76 — County scatter with $y = x$ reference line and annotations.

```
[27]: fig = px.scatter(by_county2, x='prop_dem_12', y='prop_dem_16',  
                      color_continuous_scale='bluered_r',  
                      color_continuous_midpoint=0,  
                      size='PERSONS',  
                      color='prop_dem_change',  
                      hover_data=['CNTY_NAME', 'prop_dem_change', 'PERSONS'],  
                      labels={'prop_dem_change': 'Percentage Point Change',  
                              'prop_dem_12': 'Proportion Democrat 2012',  
                              'prop_dem_16': 'Proportion Democrat 2016'},
```

```

        title='Change from 2012 to 2016',
        height=600,
        width=850)

fig.update_xaxes(range=[0, 1])
fig.update_yaxes(range=[0, 1])
fig.update_coloraxes(colorbar_tickformat='.0%')
fig.update_xaxes(tickformat='.0%')
fig.update_yaxes(tickformat='.0%')

x = np.linspace(0, 1, 10)
y = x
yx_line = go.Scatter(x=x, y=y, mode='lines',
                    line=dict(color='gray', dash='dash', width=4),
                    showlegend=False)
fig.add_trace(yx_line)

fig.add_trace(go.Scatter(
    x=[0.25], y=[0.75], mode='text', text='Got Bluer',
    textposition='middle center',
    textfont=dict(family='sans serif', size=24, color='blue'),
    showlegend=False))

fig.add_trace(go.Scatter(
    x=[0.7], y=[0.2], mode='text', text='Got Redder',
    textposition='middle center',
    textfont=dict(family='sans serif', size=24, color='red'),
    showlegend=False))

fig.show()

```

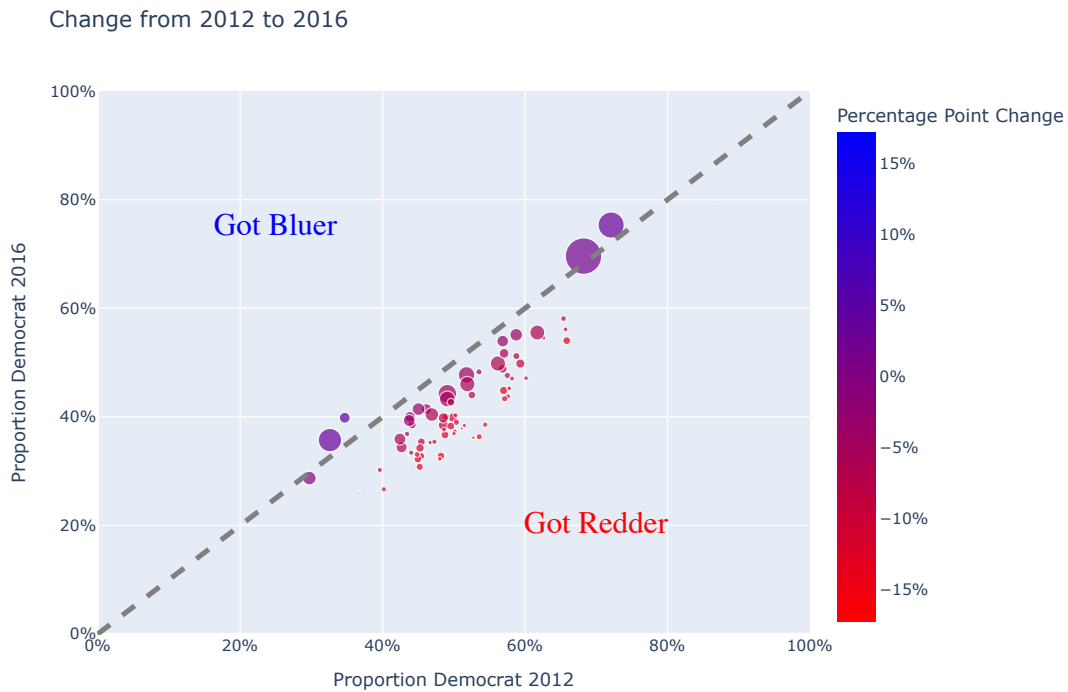


Figure 3.77 — Ward-level scatter with $y = x$ reference line.

```
[28]: pres_gdf['prop_dem_change'] = pres_gdf['prop_dem_16'] - pres_gdf['prop_dem_12']
pres_gdf.dropna(subset=['PERSONS'], inplace=True)

fig = px.scatter(pres_gdf, x='prop_dem_12', y='prop_dem_16',
                 size='PERSONS',
                 color='prop_dem_change',
                 hover_data=['CNTY_NAME'],
                 height=500,
                 width=700)

x = np.linspace(0, 1, 10)
y = x
yx_line = go.Scatter(x=x, y=y, mode='lines',
                     line=dict(color='gray', dash='dash', width=4),
                     showlegend=False)
fig.add_trace(yx_line)
fig.show()
```

