

```
In [ ]: ### Import all necessary modules
import numpy as np
!pip install geopandas;
import geopandas as gpd
from shapely.geometry import shape

from bokeh.io import output_file, show, output_notebook
from bokeh.models import ColumnDataSource, ColorBar, HoverTool
from bokeh.transform import linear_cmap
from bokeh.plotting import figure
from bokeh.palettes import RdBu10

output_notebook()
```

```
In [4]: ### Read in shapefile with election data
us_election = gpd.read_file('US48_county_election_2016.shp')

### Create helper function that converts GeoDataFrame to bokeh format
def gpd_bokeh(df):
    """Convert geometries from geopandas to bokeh format"""
    nan = float('nan')
    lons = []
    lats = []
    for i, shape in enumerate(df.geometry.values):
        if shape.geom_type == 'MultiPolygon':
            gx = []
            gy = []
            ng = len(shape.geoms) - 1
            for j, member in enumerate(shape.geoms):
                xy = np.array(list(member.exterior.coords))
                xs = xy[:,0].tolist()
                ys = xy[:,1].tolist()
                gx.extend(xs)
                gy.extend(ys)
                if j < ng:
                    gx.append(nan)
                    gy.append(nan)
            lons.append(gx)
            lats.append(gy)

        else:
            xy = np.array(list(shape.exterior.coords))
            xs = xy[:,0].tolist()
            ys = xy[:,1].tolist()
            lons.append(xs)
            lats.append(ys)

    return lons, lats
```

```
In [5]: ### Create ColumnDataSource with county name, percent of the dem votes, and percent gop vote
lons, lats = gpd_bokeh(us_election)
source = ColumnDataSource(data=dict(
    x=lons,
    y=lats,
    County = us_election['NAME'],
    Percent_Democrat = us_election['per_dem'],
    Percent_Republican = us_election['per_gop'],
))
```

```
In [6]: ### Create color map for final map that shows political lean of each county
color_mapper = linear_cmap(field_name='Percent_Republican', palette=RdBu10,
                            low=min(us_election['per_gop']),
                            high=max(us_election['per_gop']))
TOOLS = "pan,wheel_zoom,reset,hover,save"
```

```
In [7]: ### Create an interactive map that when hovered over shows county name, percent dem, and per cent gop votes
map = figure(plot_width=1000, plot_height=600, title="US: 2016 Election", tools=TOOLS, )
map.patches('x', 'y', source=source, line_color="white", line_width=0.1, color=color_mapper)

map.select_one(HoverTool).tooltips = [
    ('County Name', '@County'),
    ('Percent Democrat', '@Percent_Democrat'),
    ('Percent Republican', '@Percent_Republican')
]

color_bar = ColorBar(color_mapper=color_mapper['transform'], width=16, location=(0,0))
```

```
In [9]: show(map)
output_file("election.html")
```

In [ ]: