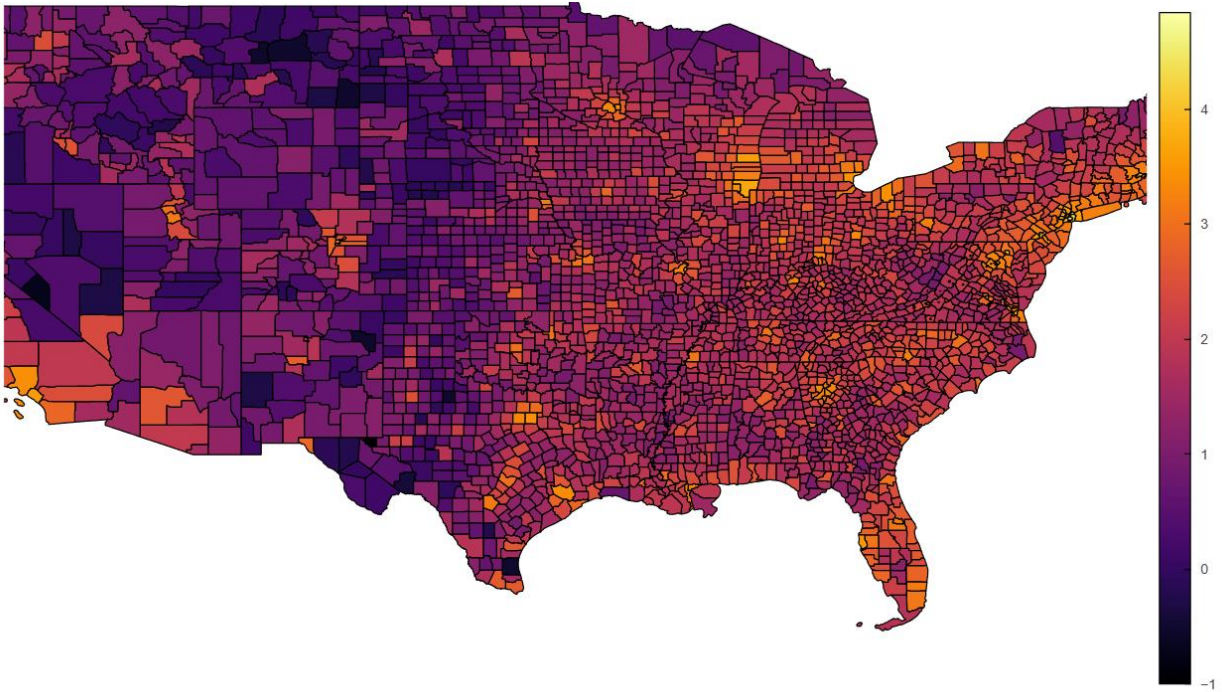


CREATING AN INTERACTIVE ZOMBIE ESCAPE MAP



Bedirhan Özkan

Barış Ergün

İlayda Saçlıoğlu

CONTENTS

| | |
|--|---|
| 1. Abstract | 3 |
| 2. Introduction | 3 |
| 2.1 Aim | 3 |
| 2.2 Scope | 3 |
| 2.3 Methodology | 3 |
| 3. Theoretical Part | 3 |
| 3.1 Choropleth Maps and Their Relevance | 3 |
| 3.2 Population Density and Safety Strategy | 4 |
| 3.3 Data Sources and Integration | 4 |
| 3.4 Visualization Libraries | 4 |
| 3.5 NumPy and the Role of Logarithmic Transformation | 4 |
| 4. Practical Part | 5 |
| 4.1 Data Loading and Preparation | 5 |
| 4.2 Cleaning and Logarithmic Transformation | 5 |
| 4.3 County Geometry Matching | 5 |
| 4.4 Comparing Data (Optional Control) | 6 |
| 4.5 Visualization and Output | 6 |
| 5. Summary | 7 |
| 6. Bibliography | 7 |

1. Abstract

To help survivors of a zombie apocalypse determine a safe escape route from Atlanta, we created an interactive choropleth map showing population density across US counties. We used real-world census data, Python data processing tools, and interactive plotting libraries to create a visual solution. The goal was to show how data science and visualization can be used to solve difficult spatial problems in emergency situations.

2. Introduction

2.1 Aim

To create an interactive US population map using population density data to help survivors avoid densely populated areas during a zombie outbreak and increase their chances of survival.

2.2 Scope

In this project, we used a CSV file with population density data for US states. The program cleaned and processed this data, then combined it with map information to create an interactive choropleth map. We saved the final map as an HTML file to open it in a web browser for easy exploration.

2.3 Methodology

We used the pandas library for data cleaning and transformation, and holoviews with a bokeh backend for plotting. Data was sourced from the US Census and Bokeh's sample datasets. The project involved joining geographical and statistical data, cleaning FIPS codes, and rendering the choropleth.

3. Theoretical Part

3.1 Choropleth Maps and Their Relevance

Choropleth maps are thematic maps that show specific geographic regions (e.g. provinces or districts) colored according to statistical data. In this project, the population density information for each district is visualized on the map. Such maps are very useful for comparing spatial data. They provide a great advantage in determining safe routes in mass escape scenarios such as zombie outbreaks, especially since they allow easy identification of sparsely populated areas.

3.2 Population Density and Safety Strategy

Scientific studies show that the chances of survival during epidemics or zombie-like disaster scenarios are higher in areas with low human density. Therefore, a map that clearly shows the density distribution of the population allows for safer routes to be planned, avoiding crowded urban centers (e.g. Atlanta). Identifying these low-density areas on the map is critical to the success of the escape plan.

3.3 Data Sources and Integration

The project used real population data from the 2010 U.S. Census and sample geographic boundary data (county boundaries) provided by the Bokeh library. In order to meaningfully connect these two data sets, a common identifier was needed: the FIPS code. This code is a five-digit numerical identifier unique to each U.S. county. This allows each county's population density to be mapped to the correct geographic area and displayed accurately on the map. The FIPS code plays a critical role in integrating information from different sources in data analysis.

3.4 Visualization Libraries

The map was built using:

- **pandas**: for loading, cleaning, and manipulating tabular data.
- **holoviews**: a high-level visualization library that simplifies map creation.
- **bokeh**: used as a backend by holoviews to generate interactive plots in HTML.

These tools were chosen for their simplicity and interactive capabilities, making the final product usable in any modern web browser without additional software.

3.5 NumPy and the Role of Logarithmic Transformation

NumPy is one of the most fundamental libraries in Python, developed for scientific and numerical calculations. In this project, it was used for logarithmic transformation operations, especially for the purpose of more meaningful visualization of population density data.

For example; While the population density is expressed in thousands in some districts, this value may remain in single digits in others. In order to perceive these large differences more easily with the human eye, the \log_{10} transformation was applied and the data was visualized in a more balanced way. Thanks to NumPy, this transformation was performed quickly and effectively on the dataset.

4. Practical Part

4.1 Data Loading and Preparation

First, population density data from the 2010 US Census was loaded using the pandas library. FIPS codes were padded with zeros to make them all 5 digits, and counties other than Alaska and Hawaii were filtered out.

```
import pandas as pd
import numpy as np
import holoviews as hv
from holoviews import opts, dim
from bokeh.sampledata.us_counties import data as counties
import hvplot.pandas # Compatibility of Holoviews with Pandas

hv.extension('bokeh')

# 1. Read population data and get FIPS codes (district codes) as string
df = pd.read_csv("census_data_popl_2010.csv", dtype={'Target Geo Id2': str})

# 2. Take the FIPS code and make it 5 digits, add 0s at the beginning
df['FIPS'] = df['Target Geo Id2'].str.zfill(5)

# 3. Incorrect data: Remove NaN ones, exclude Alaska (02xxx) and Hawaii (15xxx) because they are off the mainland
df = df[df['FIPS'].str.startswith(('02', '15'))]
```

4.2 Cleaning and Logarithmic Transformation

The density data was converted to numeric format and invalid (NaN) data was removed. For better visualization, the density data was calculated on a logarithmic scale.

```
# 4. Cleanse numeric population density data
df['Density'] = pd.to_numeric(df['Density per square mile of land area - Population'], errors='coerce')
df = df.dropna(subset=['FIPS', 'Density'])

# 5. Changes the densities logarithmically to make the difference more noticeable (optional)
df['Log_Density'] = np.log10(df['Density'].replace(0, np.nan)).fillna(0)
```

4.3 County Geometry Matching

The dataset to be visualized was created by matching the geometric district boundaries in Bokeh's sample dataset with the FIPS codes.

```
# 6. Get county geometries from Bokeh and match according to FIPS
county_shapes = []
for fips, county in counties.items():
    fips_code = f"{fips[0]:02d}{fips[1]:03d}" # FIPS: 2 digit state + 3 digit county
    if fips_code in df['FIPS'].values:
        row = df[df['FIPS'] == fips_code].iloc[0]
        county_data = {
            'lons': county['lons'],
            'lats': county['lats'],
            'name': county['name'],
            'state': county['state'],
            'FIPS': fips_code,
            'Density': row['Density'],
            'Log_Density': row['Log_Density']
        }
        county_shapes.append(county_data)
```

4.4 Comparing Data (Optional Control)

The number of records with mismatches between FIPS codes was checked.

```
# 7. Missing FIPS check (optional)
fips_in_df = set(df['FIPS'])
fips_in_counties = set(f"{f[0]:02d}{f[1]:03d}" for f in counties.keys())
missing_fips = fips_in_df - fips_in_counties
print("Missing FIPS count:", len(missing_fips))
print("Missing FIPS samples:", list(missing_fips)[:10])
```

4.5 Visualization and Output

After matching the data, an interactive map was created on a logarithmic scale with the holoviews library. The map was saved as a .html file and made available for opening via a web browser.

```
# 8. Create the map
choropleth = hv.Polygons(county_shapes, ['lons', 'lats'],
                        [('name', 'County'), 'state', 'Density', 'Log_Density'])

# 9. Visualization settings
choropleth = choropleth.opts(opts.Polygons(
    tools=['hover'],
    color=dim('Log_Density'),
    cmap='Inferno',
    colorbar=True,
    line_color='black',
    width=1000,
    height=600,
    show_grid=False,
    show_frame=False,
    xaxis=None,
    yaxis=None,
    toolbar='above',
    title='USA 2010 Log Scale Population Density Map (Excluding Alaska & Hawaii)'))

# 10. Save as HTML
hv.save(choropleth, 'us_population_density_map.html', backend='bokeh')
print("Map saved successfully as 'us_population_density_map.html'.")
```

5. Summary

The main goal of this project was to develop an interactive choropleth map simulating a data-based zombie escape scenario, and this goal was successfully achieved. Throughout the process, the practical power of Python was effectively utilized, especially in the areas of geographic data processing and interactive visualization.

Beyond producing a meaningful and useful map from real population data, this project also demonstrated the importance of presenting analysis results to the user in an interactive and accessible manner. Basic skills such as data cleaning, matching statistical data with geographic boundaries on the map, and presenting the visualization in a web environment were important parts of the process.

Thanks to the "hover" feature used on the map, users can instantly access the details of each district. Such analyses can be of great benefit not only in fictional disaster scenarios; but also in areas such as disaster management, resource planning, and public health in the real world.

6. Bibliography

- Vaughan, L. (2020). *Real-World Python*. No Starch Press.
- U.S. Census Bureau. 2010 Census Data.
- HoloViews Documentation. <http://holoviews.org/>
- Bokeh Documentation. <https://bokeh.org/>