DS210- Final Project: Mass Incarceration Analysis Project

Project Overview

This project explores mass incarceration and crime rates in the U.S. (2001–2016), focusing on relationships between incarceration rates and crime rates. By leveraging graph-based analysis, statistical modeling, and visualization, this project uncovers trends and compares Arizona and Massachusetts, two states of personal significance to me. The results address my family's concerns regarding safety during my transition from Arizona to Massachusetts for college.

Key Objectives

- Regression Modeling: Explore the relationship between incarceration and crime rates through linear and nonlinear regression.
- Comparative Analysis: Focus on crime and incarceration trends in Arizona and Massachusetts.
- Graph-Based Analysis: Use graph algorithms to identify state similarities and connectivity patterns.
- Outlier Detection and Trend Analysis: Spot anomalies and visualize nationwide trends.

Dataset Details

- Primary Dataset: crime_and_incarceration_by_state.csv
    - Contains annual crime, incarceration, and population data for each state.
    - Incarceration Rate: Calculated as prisoner_count/state_population×100,000
    - Crime Rate: Calculated as violent_crime_total/state_population×100,000

Fields:

- jurisdiction: State name (used for filtering).
- year: Year of data (used for time-series analysis).
- prisoner_count and state_population: Inputs for incarceration rate calculation.
- violent_crime_total and state_population: Inputs for crime rate calculation.

Supplemental Datasets:

- prison_custody_by_state.csv: Additional incarceration data.
- ucr_by_state.csv: Historical crime data.

Methodology

1. Data Cleaning:
    - Function: process_dataset
    - Purpose: Parse the raw dataset into valid records (CleanRecord) and log invalid ones (DirtyRecord).

Implementation:
let (clean_records, invalid_records) =
process_dataset("crime_and_incarceration_by_state.csv")?;

- Outcome: Invalid records (e.g., missing population data) are logged for transparency

2. Rate Calculations:
    - Calculated incarceration and crime rates per 100,000 residents to standardize comparisons.
3. Statistical Modeling:

    Linear Regression: To establish the relationship between incarceration rates and crime rates.

    - Function: linear_regression
    - Equation: y=0.5904x+153.4366
    - Weak positive relationship between incarceration and crime rates.

    Nonlinear Regression: To fit a quadratic model for a more accurate relationship between incarceration and crime rates

    - Function: nonlinear_regression
    - Equation: y=−0.0002x2+0.7764x+117.7482
    - Suggests diminishing returns of incarceration on crime reduction.
4. Graph-Based Analysis:

    Graph Construction:

    - Function: construct_graph
    - Connects states with similar incarceration rates.

    Degree Centrality:

    - Function: compute_degree_centrality
    - Measures the number of state-to-state connections in the similarity graph.

    K-Core Subgraph:

    - Function: compute_k_core
    - Identifies highly connected state clusters.
5. Comparative Analysis:

    Compared crime rates between Arizona and Massachusetts using:

- Function: compare_states
- Function: perform_t_test

T-Test Result:

- t-statistic: 0.03560., p-value: 0.9718
- No statistically significant difference in crime rates.

6. Visualizations:

Plotted trends, comparisons, and outliers using plotters:

- Bar charts (average incarceration rates).
- Line graphs (nationwide trends).

How to Run the Project

Prerequisites

- Install Rust and Cargo: Rust Installation Guide.
- Ensure the required input files are in the project directory.

Steps

1. Clone the repository:
   git clone https://github.com/ivettealcantar/DS210Project.git
   cd DS210Project

2. Build and run the project:

   cargo build

   cargo run

3. Outputs are saved in the output directory.

Output Files

- rates.png:Bar chart of average incarceration/crime rates.
- nonlinear_regression.png: Quadratic regression visualization.
- national_averages.png: Nationwide trends (2001–2016).
- arizona_trends_over_time.png: Arizona-specific trends.
- massachusetts_trends_over_time.png: Massachusetts-specific trends.
- degree_centrality_chart.png: Degree centrality of states.
- az_mass_crime_rates_comparison.png: Arizona vs. Massachusetts comparison.
- Graph.dot: Graph data for further visualization.

nonlinear_regression.png

national_averages.png

Key Results

1. National Trends:
    - Nationwide incarceration and crime rates generally declined from 2001 to 2016.
2. State Comparisons:
    - Arizona and Massachusetts show no statistically significant difference in crime rates (p=0.97p = 0.97p=0.97).
3. Graph Analysis:
    - Highly connected states: California, Colorado, and Illinois.
    - Outliers: Alaska (high crime rates in 2011–2014).
4. Centrality States:
    a. High Degree Centrality States:
    - States such as Illinois, Indiana, California, Montana, and Vermont show the highest degree centrality values (around 2000).
    - These states have strong interconnections with other states, indicating similar incarceration rates compared to many others.
    b. Medium Degree Centrality States:
    - States like Arizona, Massachusetts, Maine, and South Carolina have moderate degree centrality values (~1000–1500).
    - These states exhibit fewer but still notable connections with other states.
    c. Low Degree Centrality States:
    - States such as Alaska, Delaware, Oklahoma, and Texas have the lowest degree centrality values, indicating fewer connections.
    - This suggests that their incarceration rates differ significantly from most other states.

Other code used

1. Checking Dataset Existence

```
if !std::path::Path::new("crime_and_incarceration_by_state.csv").exists() {
    eprintln!("Dataset file not found. Please ensure the file is present.");
    return Ok(());
```

```
}
```

- Ensures required files are present before running.

2. Linear Regression

```
let slope = x.iter().zip(y.iter()).map(|(xi, yi)| (xi - mean_x) * (yi - mean_y)).sum::<f32>()
   / x.iter().map(|xi| (xi - mean_x).powi(2)).sum::<f32>();
```

- Fits a line to describe the relationship between incarceration and crime rates.

3. Graph Construction

```
for i in 0..records.len() {
   for j in i + 1..records.len() {
      let rate_diff = (state1.incarceration_rate - state2.incarceration_rate).abs();
      if rate_diff < 50.0 {
         graph.add_edge(idx1, idx2, rate_diff);
      }
   }
}
```

- Builds a similarity graph connecting states with close incarceration rates.