Visualizing V😢ter Suppression

Andres De La Fuente
Anmol Srivastava
Juan Solorio

Background

Our goals:

- draw attention to inhibited turnout
- contextualize with trends, contributing factors
- provide user-friendly visuals

The programming challenges:

- design dashboard with many variables
- add intuitive interactivity
- fit malformed data into the development process

Data Used

US Census Bureau

- *annual voting, registration data*
- *accompanying demographic data*

American Civil Liberties Union (ACLU)

- *latest ID legislation data*
- *latest felony disenfranchisement data*

Data Used

US Census Bureau

- *publicly available*
- *large online data-bank*
- *annual updates*

American Civil Liberties Union (ACLU)

- *publicly available*
- *online articles*

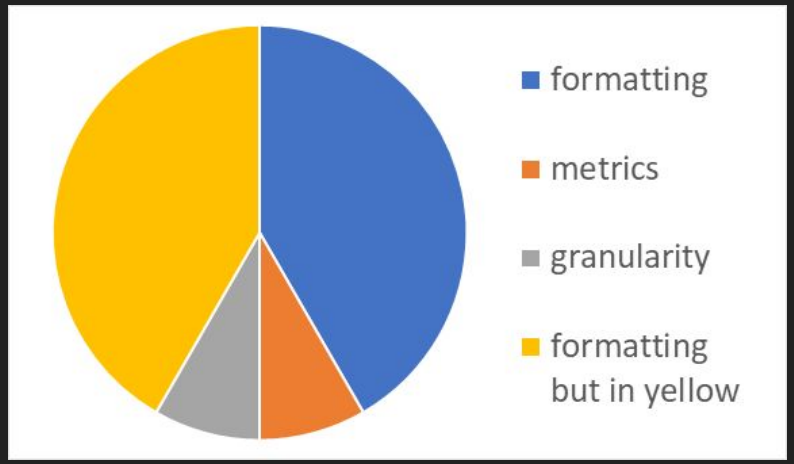
Data Used

US Census Bureau

- *data format*
- *inconsistent metrics*
- *inconsistent granularity*

American Civil Liberties Union (ACLU)

- *lack of data for multiple years*



Use Case #1

The user explores changes in turnout over time, by:

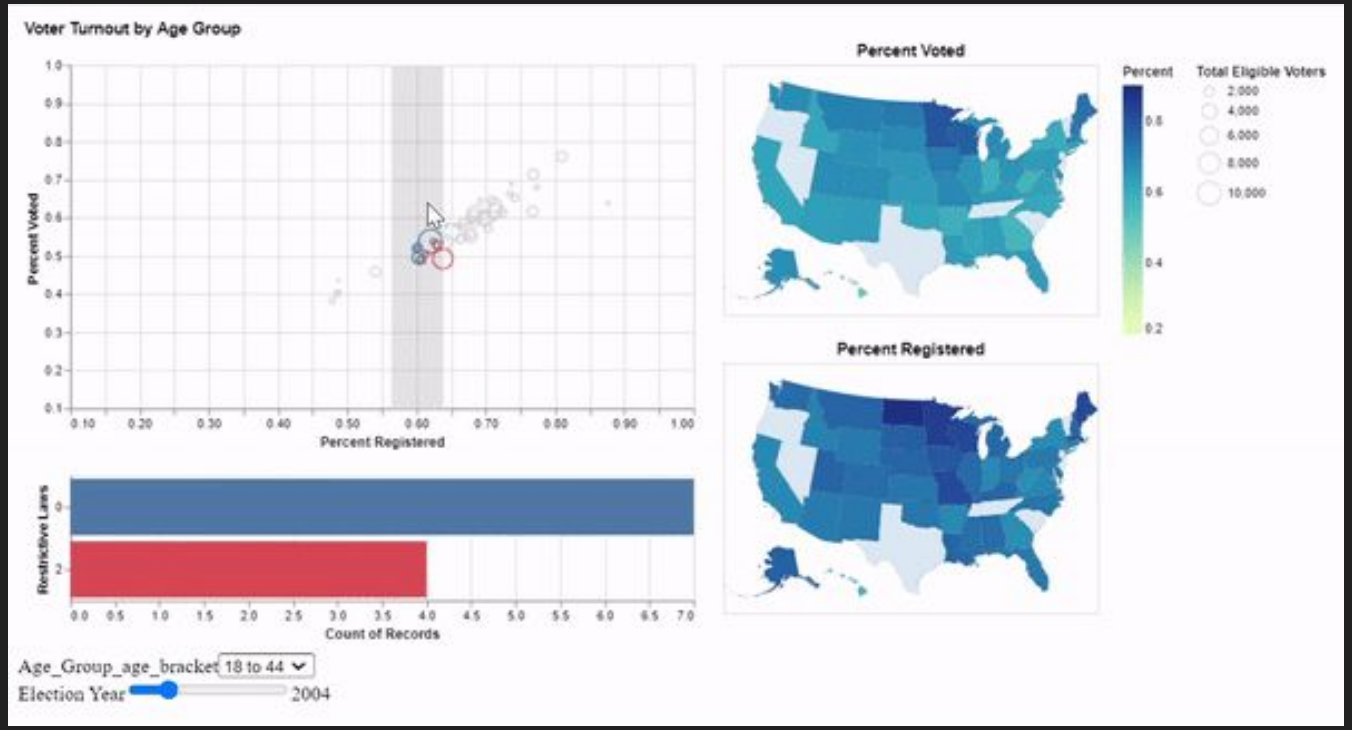
- using the slider to change the visualization data by year
- viewing the updated plots
- Explore different relations in the voting patterns

Use Case #2:

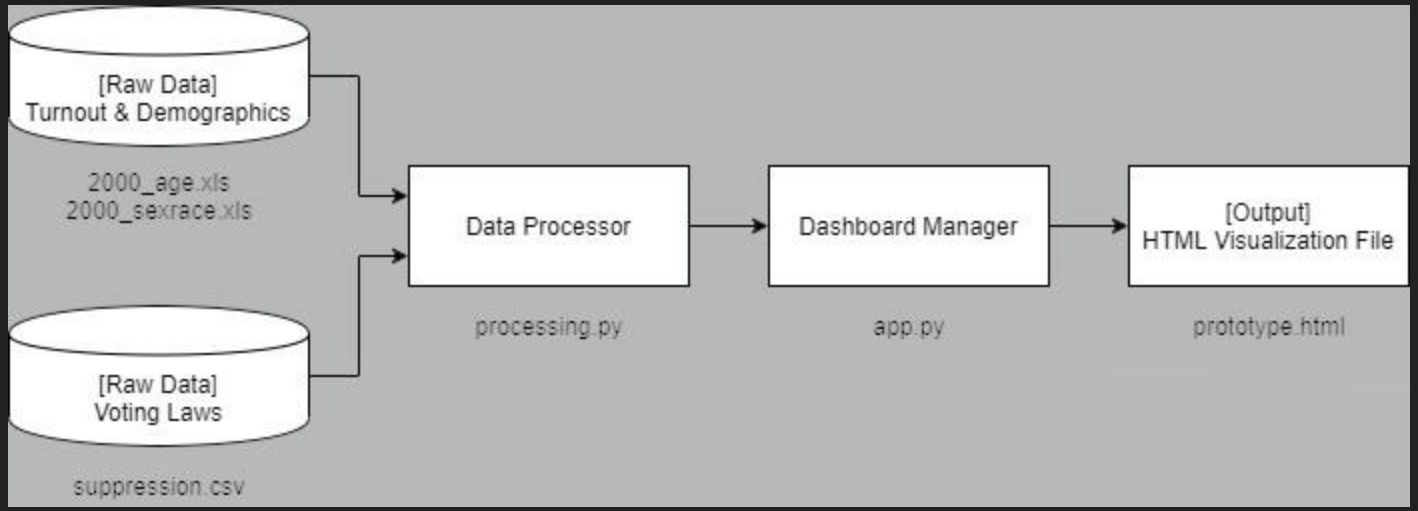
The user explores state-wise differences in turnout, for a fixed year, by:

- keeping the slider fixed for a specific election
- using the select, drop-down, and hover features for different demographics
- viewing updated maps, and plots for turnout and disenfranchisement

Demonstration



Design


[Raw Data]
Turnout & Demographics
2000_age.xls
2000_sexrace.xls
[Raw Data]
Voting Laws
suppression.csv
Data Processor
processing.py
Dashboard Manager
app.py
[Output]
HTML Visualization File
prototype.html

Project Structure

```
.
├── docs/
│   ├── pylint_scores/
│   ├── component-specifications.md
│   ├── functional-specifications.md
│   └── technology-review.pdf*
├── examples/
│   └── Dashboard Tutorial.ipynb
├── voter_suppression_analysis/
│   ├── data/
│   │   ├── clean/
│   │   ├── raw/
│   │   ├── samples/
│   │   └── README.md
│   ├── figures/
│   │   ├── chart_prototype.html
│   │   └── dashboard.html
│   ├── tests/
│   │   ├── test_generate.py*
│   │   └── test_processing.py*
│   ├── generate.py*
│   └── processing.py*
├── LICENSE*
├── README.md
├── environment.yml
├── pytest.ini
└── setup.py*
```

repo

Outlook

Lessons learned:

- data considerations
- live feedback between dashboard and backend is arduous
- established visualization tools may be better

Future work:

- wrapper to integrate outputs with other projects
- enhance map interactivity
- explore more variables and deeper analyses