Neighborhood Traffic Flow

Kareha Agesa
Siqi Cheng
Becca Esquenazi
Kelsey Maass

https://github.com/Neighborhood-Traffic-Flow/neighborhoodtrafficflow

Background

- When moving to a new city/neighborhood, it is difficult to know the area's traffic flow
  - Sites such as Zillow, Craigslist, etc., do not provide information about whether property is located near a busy street
- This information may help movers make a decision on whether or not to buy a house in a particular location
- **We created an interactive mapping tool that provides users with information about vehicle traffic, speed limits, and road types for neighborhoods in Seattle**
  - Users can to filter by neighborhood to discover traffic flow, speed limits, and arterial classifications
  - Users can view historical data about traffic flow
  - Users can compare neighborhoods statistics to the rest of the city

Data used

- Zillow Neighborhoods
  - GIS Seattle neighborhood data from Zillow
- Seattle street data
  - City of Seattle Open Portal GIS street data with street names, speed limits, and arterial classifications
- Traffic flow counts
  - City of Seattle Open Portal GIS traffic flow data with street names and average weekday traffic flow counts

Limitations:

- Some years reported different types of traffic flow
- Some years had different variables/systems to identify/group road segments

Use cases

- Users who just moved into a new area want to see the traffic flow in that specific neighborhood for the past few years
- Users have located homes of interest and they want to see the road types and historical traffic flows in those neighborhood to make better decisions
- Users have young children and care about the speed limit in their neighbourhood
- Users have determined homes of interest, and want to see how the traffic flow in their neighborhood compares to other areas in Seattle

Demo

Dashboard: http://127.0.0.1:8050/

Design

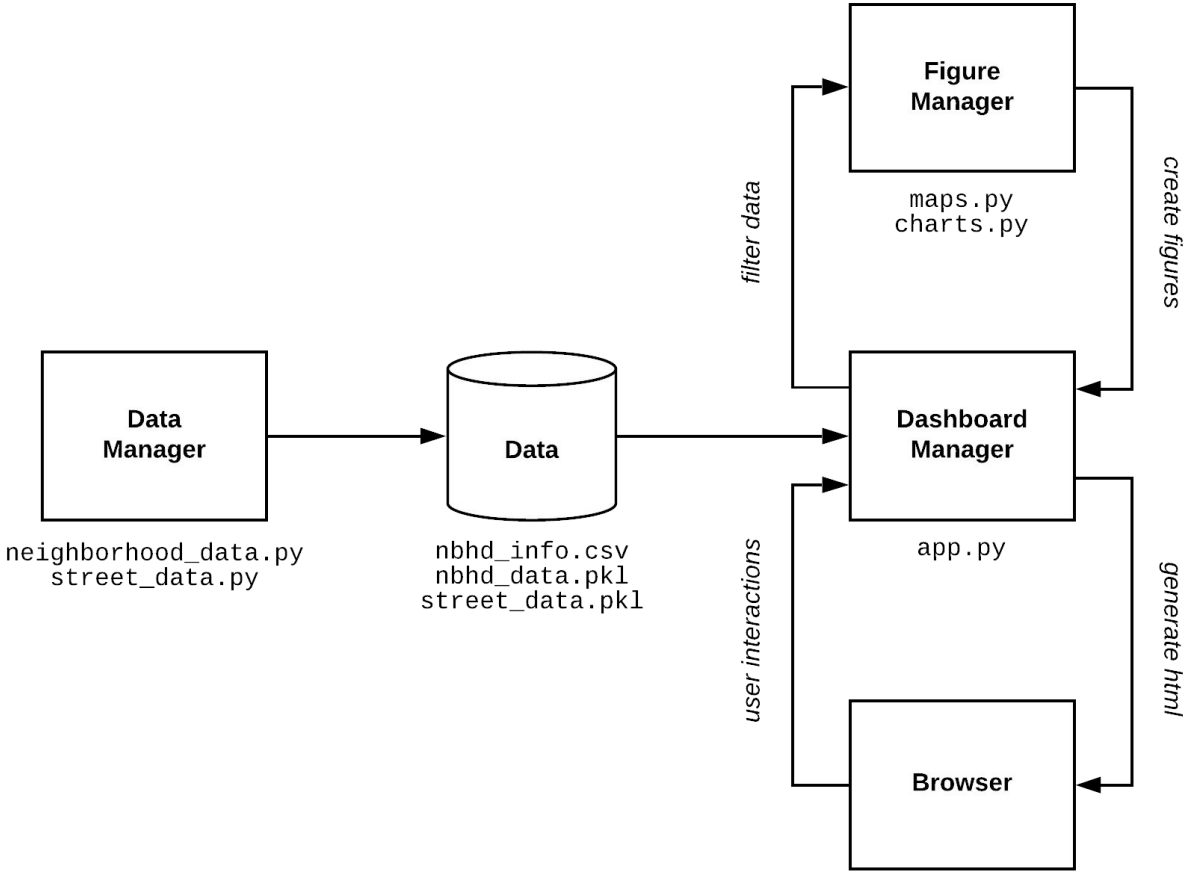
Figure Manager
maps.py
charts.py
filter data
create figures
Data Manager
neighborhood_data.py
street_data.py
Data
nbhd_info.csv
nbhd_data.pkl
street_data.pkl
Dashboard Manager
app.py
user interactions
generate html
Browser

Project structure

```
neighborhoodtrafficflow/
     |- docs/
     |- examples/
     |- neighborhoodtrafficflow/
     |- .gitignore
     |- .travis.yml
     |- LICENSE
     |- README.md
     |- environment.yml
     |- pytest.ini
     |- setup.py
```

Project structure

```
neighborhoodtrafficflow/
     |- docs/
     |- examples/
     |- neighborhoodtrafficflow/ ──────────► |- assets/
     |- .gitignore                           |- data/
     |- .travis.yml                          |- figures/
     |- LICENSE                              |- tests/
     |- README.md                            |- app.py
     |- environment.yml
     |- pytest.ini
     |- setup.py
```

Project structure

```
neighborhoodtrafficflow/
     |- docs/
     |- examples/
     |- neighborhoodtrafficflow/ ──────────► |- assets/
     |- .gitignore                           |- data/
     |- .travis.yml                          |- figures/
     |- LICENSE                              |- tests/
     |- README.md                            |- app.py
     |- environment.yml
     |- pytest.ini
     |- setup.py
```

CSS style sheets

Project structure

```
neighborhoodtrafficflow/
     |- docs/
     |- examples/
     |- neighborhoodtrafficflow/ ──────────► |- assets/
     |- .gitignore                           |- data/
     |- .travis.yml                          |- figures/
     |- LICENSE                              |- tests/
     |- README.md                            |- app.py
     |- environment.yml
     |- pytest.ini
     |- setup.py
```

Data manager

Project structure

```
neighborhoodtrafficflow/
     |- docs/
     |- examples/
     |- neighborhoodtrafficflow/ ──────────► |- assets/
     |- .gitignore                           |- data/
     |- .travis.yml                          |- figures/
     |- LICENSE                              |- tests/
     |- README.md                            |- app.py
     |- environment.yml
     |- pytest.ini
     |- setup.py
```

Figure manager

Project structure

```
neighborhoodtrafficflow/
     |- docs/
     |- examples/
     |- neighborhoodtrafficflow/ ──────────► |- assets/
     |- .gitignore                           |- data/
     |- .travis.yml                          |- figures/
     |- LICENSE                              |- tests/       Tests
     |- README.md                            |- app.py
     |- environment.yml
     |- pytest.ini
     |- setup.py
```

Project structure

```
neighborhoodtrafficflow/
     |- docs/
     |- examples/
     |- neighborhoodtrafficflow/ ──────────► |- assets/
     |- .gitignore                           |- data/
     |- .travis.yml                          |- figures/
     |- LICENSE                              |- tests/
     |- README.md                            |- app.py        Dashboard manager
     |- environment.yml
     |- pytest.ini
     |- setup.py
```

Lessons learned

- Tradeoff in efficiency vs. visualization features
- Testing is more useful when you do it from the start!
- Problem solving in merging datasets when there are differences in how the city reported/organized street data over the years
- Learned to work with Dash, Plotly, and geographic data

Future Work

- Merge datasets that have similarly reported statistics and facts
- Auto-update dashboard every time new data is published on seattle.gov website
- Incorporate different data sources
- Different visualizations/charts