Equitable Fire Hydrant Placement

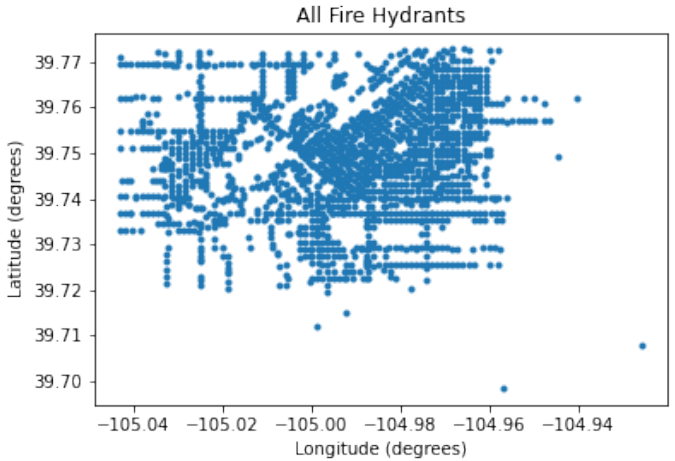
Johnathan Rhyne

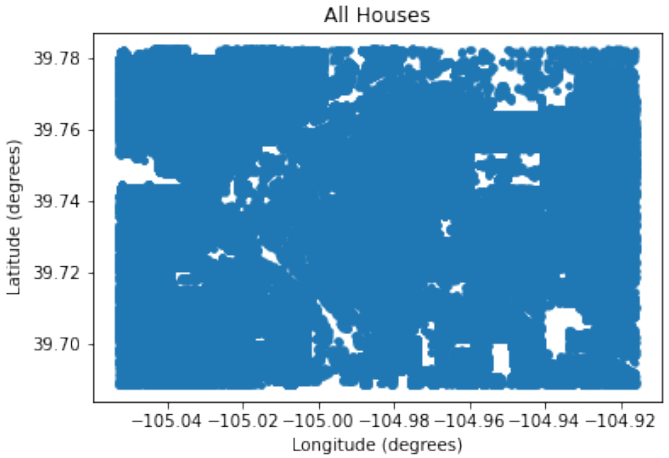
Motivation

- Building new housing incurs more costs than just the construction of the building itself
- Safety of inhabitants is important
- Can affect home insurance rates

Limitations

- A LOT of buildings in our data set
- Hard to access official data
- Sparse crowd sourced data outside Denver-Metro area





Design Decisions

- Precompute all distances
- Close proximity means euclidean distance is sufficient

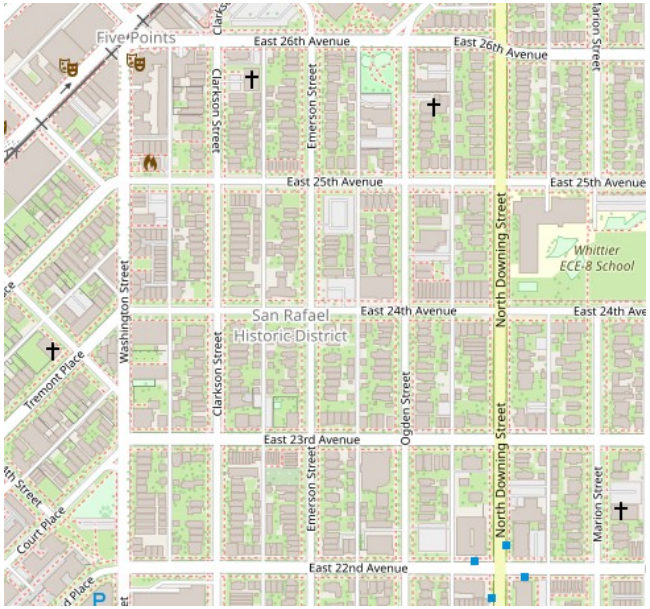
Clustering Program (LP with Integer Solution)

File: **Clustering.mod**

```
param numHydrant integer >= 0;
param numHouses integer >= 0;
set hydrants := 1..numHydrant;
set houses := 1..numHouses;
param distances{houses,hydrants} >= 0;

# Determines if we have a particular house i to a hydrant j
var choose {houses,hydrants} >= 0;
var cost{hydrants} >= 0; # Exists due to a hacky way of me creating data files

minimize distanceFunc: sum {i in houses, j in hydrants} distances[i,j] * choose[i,j];

subject to one_hydrant_per_house {i in houses}:
    sum {j in hydrants} choose[i,j] = 1;

```

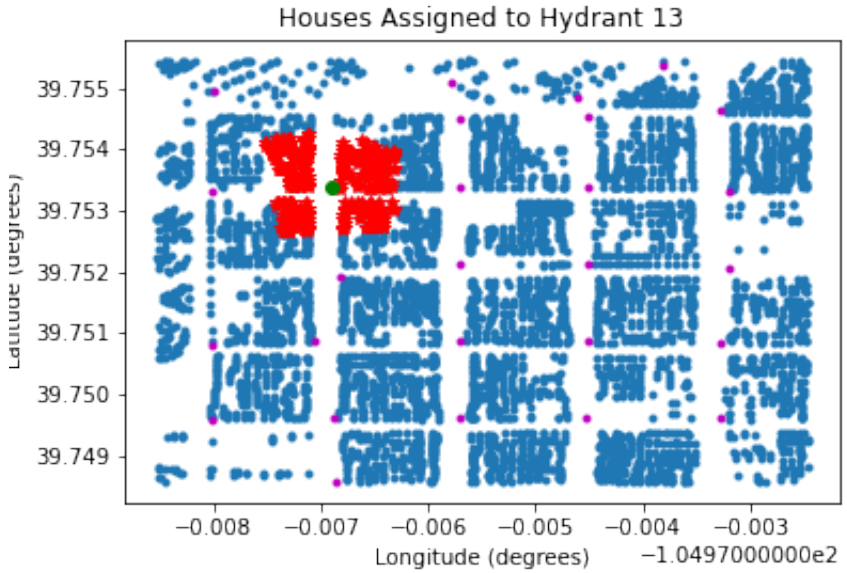
Region Selection Methods

- Selecting varying size regions distributed throughout our data set
- Could be refined based on other factors



Current State

- Assigning every house to its closest hydrant



Facility Location Program

File: **Location.mod**

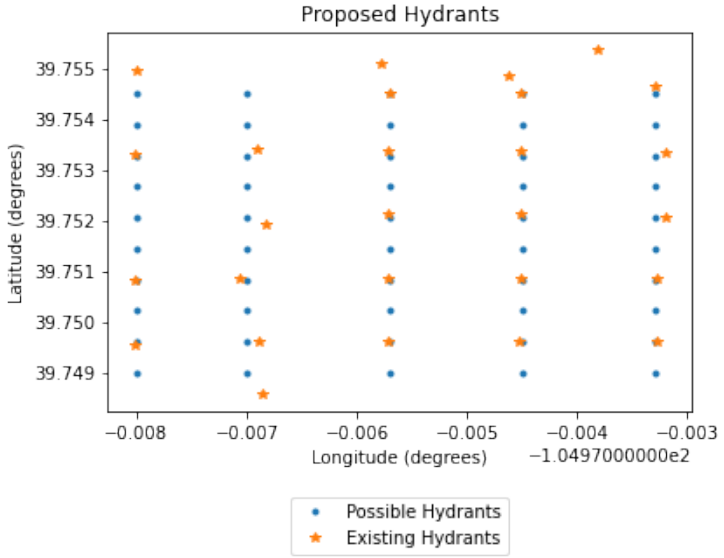
```
param numHydrant integer >= 0;
param numHouses integer >= 0;
set hydrants := 1..numHydrant;
set houses := 1..numHouses;
param distances{houses,hydrants} >= 0;
param cost{hydrants} >= 0;

# Determines if we have a particular house i to a hydrant j
var choose {houses,hydrants} >= 0;
# Determines if we use a particular hydrant
var makeHydrant {hydrants} binary;

minimize distanceFunc: sum {i in houses, j in hydrants} distances[i,j] * choose[i,j] + sum {j in hydrants} 10000 * makeHydrant[j] * cost[j];

subject to one_hydrant_per_house {i in houses}:
    sum {j in hydrants} choose[i,j] = 1;

subject to needMadeFirst { i in houses, j in hydrants }:
    choose[i,j] <= makeHydrant[j];

#subject to maxPerHydrant { j in hydrants }:
#    sum { i in houses } choose[i,j] <= 200
```

Homes Per Hydrant

- Needs refinement
  - Empirical evidence on choice desired

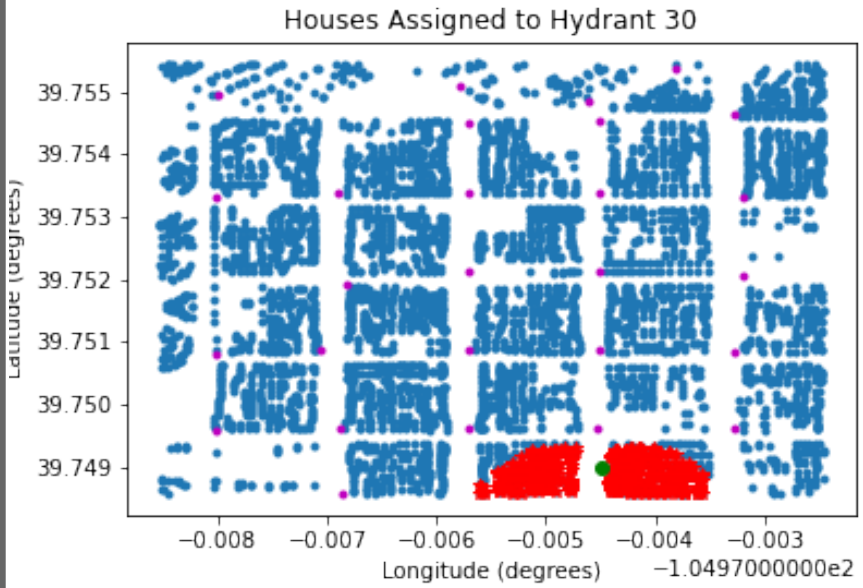
Methods

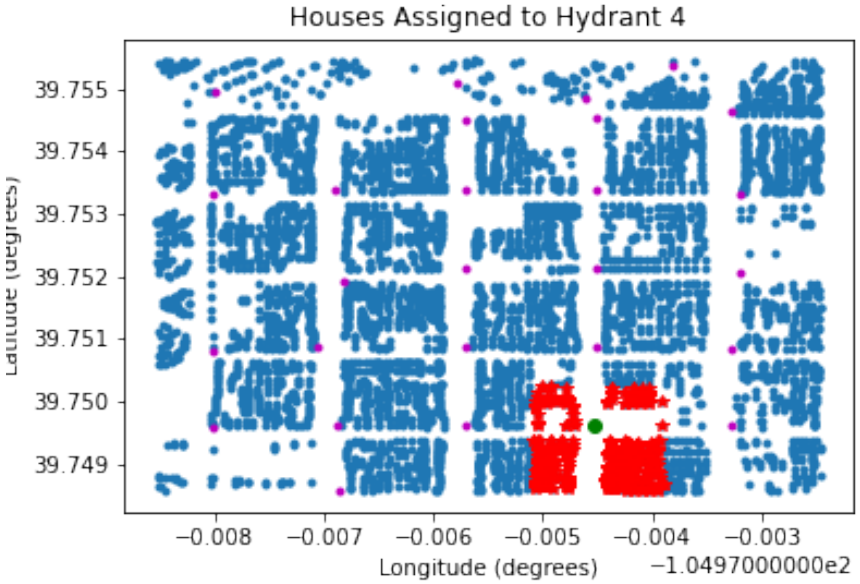
- In each region, determine placement of hydrants to minimize the distance from each building to its hydrant.
- Penalize creating new hydrants



Proposed Placement

- Very few differences in our testing





Analysis

- No real difference between “ideal” and actual
- Consider different penalization for adding new hydrants

Future work

- Better data: Clear distinction for homes and businesses
- Look at regions with clear disparities in socioeconomic status
- Better measure of equitability: Treat higher density apartments as more expensive than homes due to more people
- Consider fire risk

Citations

- Denver Fire code
- Overpass Turbo (Used for getting our data)
- Insurance rate increases
- Source Code