

Assignment 2

CPLN 670 - Geospatial Software Design

Professor: Dr. Dana Tomlin

Student: Zixi Liu

“The Best Tract to Build a Private Hospital in Connecticut”

According to The 500 Cities Project, a collaboration by the CDC, the CDC Foundation, and the Robert Wood Johnson Foundation (RWJF), estimates of different health indicators in three categories of measurement—preventive care, health outcomes, and unhealthy behaviors—at city- and census tract-levels enables researchers to understand health, neighborhood by neighborhood. Here in this project, we are mainly using factors such as population density, median household income, percentage of 50-70 years old people, percentage of 70-90 years old people and location of large cities to determine the best tract to build a private hospital in Connecticut.

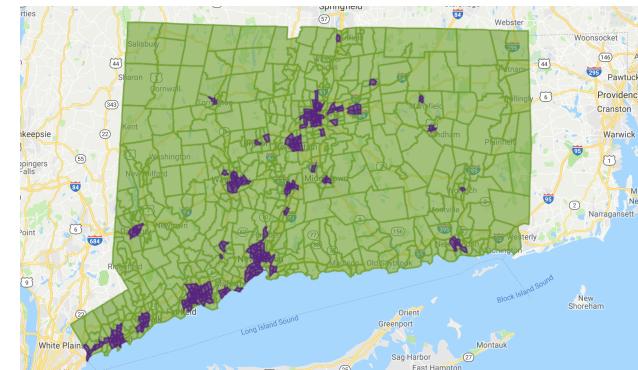
```
//Zoom to Connecticut
var Connecticut = ee.FeatureCollection('ft:1xa2PvKTf7ynyAAEXEeHoltriaHFkyFJpvd74BLc6');
Map.centerObject( Connecticut, 9 );
Map.addLayer( Connecticut, { color: '72A231' }, 'Tracts');
print(Connecticut);
```

Assignment 2

We first filter out areas where our value of interest is 0.

```
//filter out tracts where have values of 0.  
//inspired by past submission by chengcheng qiu.  
var filterConnecticut = Connecticut.filterMetadata('AREA', 'greater_than', 0);  
var filterConnecticut = Connecticut.filterMetadata('Med_Inc', 'greater_than', 0);  
var filterConnecticut = Connecticut.filterMetadata('POP', 'greater_than', 0);  
var filterConnecticut = Connecticut.filterMetadata('Pct_DU5070', 'greater_than', 0);  
var filterConnecticut = Connecticut.filterMetadata('Pct_DU7090', 'greater_than', 0);
```

Step 1. We find the census tracts that have higher population density.

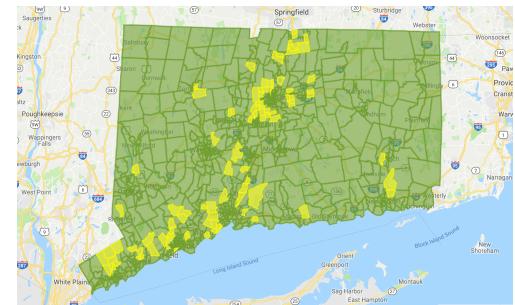


```
//Find the census tracts with higher population density.  
var ConnecticutAddFeature = function(feature) {  
  var PopulationDensity = ee.Number(feature.get('POP')).divide(ee.Number(feature.get('AREA')));  
  return feature.set({'PopulationDensity': PopulationDensity});};  
var ConnecticutAddFeature = ee.FeatureCollection(filterConnecticut.map(ConnecticutAddFeature));  
print(ConnecticutAddFeature.getInfo());  
  
///Filter all the census tracts whose population density higher than average density.  
var meanPopDensity = ee.Number(ConnecticutAddFeature.aggregate_mean('PopulationDensity'));  
var highPopDensity = ConnecticutAddFeature.filterMetadata('PopulationDensity', 'greater_than', meanPopDensity);  
var sortPopDensity = highPopDensity.sort('PopulationDensity', false);  
var topPopDensity = sortPopDensity.limit(100);  
Map.addLayer(highPopDensity,{color:'5B2782'},'Tracts with Higher Population Density');
```

Assignment 2

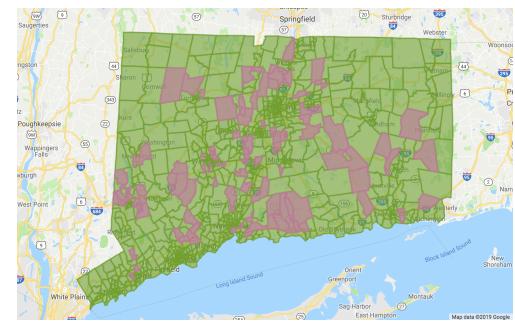
Step 2. We then find the tracts that have higher percentage of 50-70 years old people.

```
//Find the census tracts with higher percentage of 50-70yrs old people.  
var mean5070 = ee.Number(filterConnecticut.aggregate_mean('Pct_DU5070'));  
var high5070 = ConnecticutAddFeature.filterMetadata('Pct_DU5070', 'greater_than',mean5070);  
var sort5070 = high5070.sort('Pct_DU5070',false);  
var top5070 = sort5070.limit(100);  
Map.addLayer(top5070,{color:'F9F91A'},'Tracts with Higher Percentage of 50-70 Yrs Old People');
```



Then, we find the tracts that have higher percentage of 70-90 years old people.

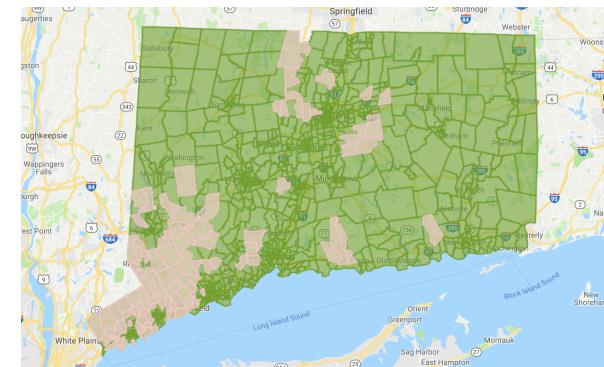
```
//Find the census tracts with higher percentage of 70-90yrs old people.  
var mean7090 = ee.Number(filterConnecticut.aggregate_mean('Pct_DU7090'));  
var high7090 = ConnecticutAddFeature.filterMetadata('Pct_DU7090', 'greater_than',mean7090);  
var sort7090 = high7090.sort('Pct_DU7090',false);  
var top7090 = sort7090.limit(100);  
Map.addLayer(top7090,{color:'C27BA0'},'Tracts with Higher Percentage of 70-90 Yrs Old People');
```



Assignment 2

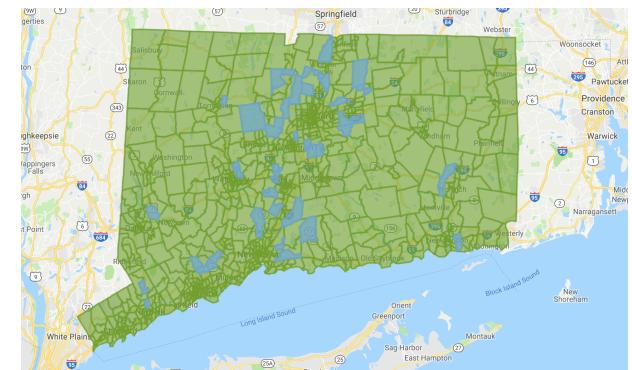
Step 3. We find the tracts that have higher median household income.

```
//Find the census tracts with higher household income.  
var MedianIncome = ee.Number(filterConnecticut.aggregate_mean('Med_Inc'));  
var highMedianIncome = Connecticut.filterMetadata('Med_Inc', 'greater_than', MedianIncome);  
var sortMedianIncome = highMedianIncome.sort('Med_Inc', false);  
var topMedianIncome = sortMedianIncome.limit(100);  
Map.addLayer(topMedianIncome, {color: 'F4CCCC'}, 'Tracts with Higher Median Income');
```



Step 4. Then we select the intersection with higher percentage of 50-70 and 70-90 years old people.

```
//Select the areas with higher percentage of both 50-70yrs and 70-90yrs old people.  
var intersection5090 = top7090.filterBounds(top5070);  
Map.addLayer(intersection5090, {color: '6FA8DC'}, 'Tracts with Higher Percentage of Both 50-70yrs and 70-90yrs Old People.');
```



Assignment 2

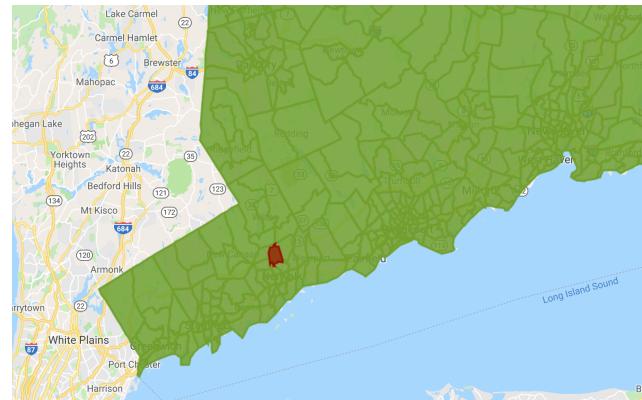
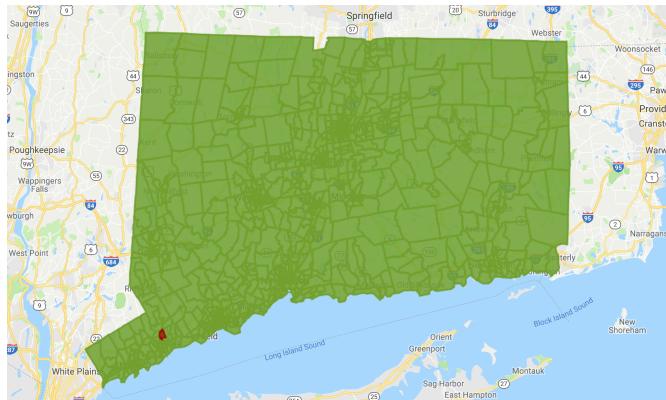
We then intersect the tracts with higher population density, higher percentage of old people and higher median household income.

```
var intersectionPOP_OLD = intersection5090.filterBounds(topPopDensity);
//Map.addLayer(intersectionPOP_OLD,{color:'990000'},'Tracts with Higher Percentage of Both 50-70yrs and
print(intersectionPOP_OLD);

var intersection = intersectionPOP_OLD.filterBounds(topMedianIncome);
Map.addLayer(intersection,{color:'990000'},'Tracts with Higher Percentage of Both 50-70yrs and 70-90yrs
print(intersection);
var choice = intersection.limit(1);
print(choice);
//Map.addLayer(choice,{color:'FF00FF'},'Best Tract');
```

```
▼ FeatureCollection (1 element, 38 columns)
  type: FeatureCollection
  ▷ columns: Object (38 properties)
    ▷ features: List (1 element)
      ▷ 0: Feature 2701 (Polygon, 38 properties)
        ▷ properties: Object (2 properties)
```

We finally arrive at one tract.



Assignment 2

We want to double check with the location of the five largest cities in Connecticut by creating buffers around cities.

```
var Bridgeport = ee.Geometry.Point(-73.195557,41.186390).centroid().buffer(5000).dissolve();
var NewHaven = ee.Geometry.Point(-72.928160,41.308150).centroid().buffer(5000).dissolve();
var Stamford = ee.Geometry.Point(-73.536216,41.054082).centroid().buffer(5000).dissolve();
var Hartford = ee.Geometry.Point(-72.677099,41.78007).centroid().buffer(5000).dissolve();
var Waterbury = ee.Geometry.Point(-72.927507,41.365709).centroid().buffer(5000).dissolve();

print(Bridgeport);
print(NewHaven);
print(Stamford);
print(Hartford);
print(Waterbury);

//Locate the five largest cities in Connecticut
Map.addLayer(Bridgeport,{color:'FFFF00'},'Bridgeport');
Map.addLayer(NewHaven,{color:'FFFF00'},'NewHaven');
Map.addLayer(Stamford,{color:'FFFF00'},'Stamford');
Map.addLayer(Hartford,{color:'FFFF00'},'Hartford');
Map.addLayer(Waterbury,{color:'FFFF00'},'Waterbury');
```

The area we selected is close to four of the largest cities and has higher population density, more seniors, and higher household income and thus tract 2701 is the best tract to build a private hospital in Connecticut.

