Assignment 03

CPLN 670 - Geospatial Software Design
Professor: Dr. Dana Tomlin
Student: Zixi Liu

"Score the Census Tracts of Connecticut on the Basis of Their Likeliness to Support Gun Control"

After conducting research on the factors that may affect people's likeliness to support gun control, we examine mainly three factors – demographic factor, economic factor, and social factor – to score the census tracts of Connecticut on the Basis of their likeliness to support gun control.

The criteria we used is listed as below:

i. Demographic Factor.
- Family Density; higher family density tend to support gun control.
- Population Density; higher population density tend to support gun control.
- Urban Area or Suburban Area; urban areas tend to support gun control.

ii. Economic Factor.
- Household Income; people with lower household income tend to support gun control.
- House Value; lower house value residents tend to support gun control.

iii. Social Factor.
- African American Percentage of Population; African Americans tend to support gun control.
- Population Percentage Below Poverty Level; people below poverty level tend to support gun control.
- Multi-Family Dwelling Unit; people living in multi-family are more likely to support gun control than single-family.

Link to code: https://code.earthengine.google.com/628e45d937abf443f560e38f5500aa52

Assignment 03

Step one. Zoom to the layer of Connecticut and pre-process data.

```
//Add the layer of Connecticut
var Connecticut = ee.FeatureCollection("ft:1xa2PvKTf7ynyAAEXEeHoltriaHFkyFJpvd74BLc6");
print(Connecticut)
Map.addLayer( Connecticut, { color: 'D9EAD3' }, 'Connecticut Census Tract' );
Map.centerObject( Connecticut, 8.5 );

//Data preprocessing
var Connecticut = Connecticut.filterMetadata('Families','greater_than',0);
var Connecticut = Connecticut.filterMetadata('Med_Inc','greater_than',0);
var Connecticut = Connecticut.filterMetadata('Med_Value','greater_than',0);
var Connecticut = Connecticut.filterMetadata('POP','greater_than',0);
var Connecticut = Connecticut.filterMetadata('Pct_AfAm','greater_than',0);
var Connecticut = Connecticut.filterMetadata('Pct_BlPov','greater_than',0);
var Connecticut = Connecticut.filterMetadata('Pct_MF_DU','greater_than',0);
var Connecticut = Connecticut.filterMetadata('Pct_SF_DU','greater_than',0);

print(Connecticut.getInfo());
```

Figure: Layer of Connecticut

Step two. Compute Demographic Score

```
//Demographic score;
//Create functions to calculate population density and family density.
function pop_density(feature)
{
  var area = ee.Number(feature.get('AREA'));
  var pd = ee.Number(feature.get('POP')).divide(area);
  return feature.set({'pop_density': pd});
}

var popDensity = ee.FeatureCollection(Connecticut.map(pop_density));
print(popDensity.getInfo());

function fam_density(feature)
{
  var area = ee.Number(feature.get('AREA'));
  var fd = ee.Number(feature.get('Families')).divide(area);
  return feature.set({'fam_density':fd});
}

var famDensity = ee.FeatureCollection(popDensity.map(fam_density));
print(famDensity.getInfo());
```

Link to code: https://code.earthengine.google.com/628e45d937abf443f560e38f5500aa52

Assignment 03

```
// Define a spatial filter, with distance 10 km.
var distFilter = ee.Filter.withinDistance({
  distance: 10000,
  leftField: '.geo',
  rightField: '.geo',
  maxError: 10
});

var simpleJoin = ee.Join.simple();

//add big cities
// Make a list of Features.
var features = [
  ee.Feature(ee.Geometry.Point(-73.195557,41.186390), {name: 'Bridgeport'}),
  ee.Feature(ee.Geometry.Point(-72.928160,41.308150), {name: 'NewHaven'}),
  ee.Feature(ee.Geometry.Point(-73.536216,41.054082), {name: 'Stamford'}),
  ee.Feature(ee.Geometry.Point(-72.677099,41.78007), {name: 'Hartford'}),
  ee.Feature(ee.Geometry.Point(-72.927507,41.365709), {name: 'Waterbury'}),
  ee.Feature(ee.Geometry.Point(-73.4079,41.1176), {name: 'Norwalk'}),
  ee.Feature(ee.Geometry.Point(-73.471416,41.376242), {name: 'Danbury'}),
  ee.Feature(ee.Geometry.Point(-72.778391,41.666049), {name: 'New_Britain'}),
];

var bigCities = ee.FeatureCollection(features);
print(bigCities.getInfo());
```

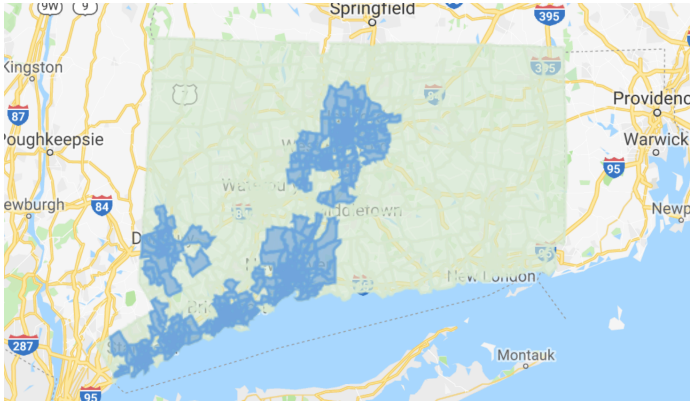


Figure: Urban Areas in Connecticut

```
// Apply the join.
var spatialJoined = simpleJoin.apply(famDensity, bigCities, distFilter);
print(spatialJoined);

Map.addLayer(spatialJoined,{color:'6FA8DC'},'Urban Areas');

//Define a function to compute demographic score;
var famDen_minmax = spatialJoined.reduceColumns(ee.Reducer.minMax(),['fam_density']);
var popDen_minmax = spatialJoined.reduceColumns(ee.Reducer.minMax(),['pop_density']);
var fam_max = ee.Number(famDen_minmax.get('max')).log();
var fam_min = ee.Number(famDen_minmax.get('min')).log();
var pop_max = ee.Number(popDen_minmax.get('max')).log();
var pop_min = ee.Number(popDen_minmax.get('min')).log();
var fd_diff = fam_max.subtract(fam_min);
var pd_diff = pop_max.subtract(pop_min);
print(famDen_minmax);
print(popDen_minmax);
```

Link to code: https://code.earthengine.google.com/628e45d937abf443f560e38f5500aa52

Assignment 03

```
function demo_score(feature)
{
  var fd = ee.Number(feature.get('fam_density')).log();
  var pd = ee.Number(feature.get('pop_density')).log();
  var geo = feature.geometry();
  var urban_score = 0;
  var demoScore = 0;
  var fd_score = fd.subtract(fam_min).multiply(20).divide(fd_diff);
  var pd_score = pd.subtract(pop_min).multiply(20).divide(pd_diff);
  if (geo.contains(spatialJoined.geometry()))
  {
    urban_score = 60;
  }
  else
  {
    urban_score = 0;
  }
  demoScore = fd_score.add(pd_score).add(urban_score);
  return feature.set({'demoScore':demoScore});
}
var demoScore = ee.FeatureCollection(spatialJoined.map(demo_score));
print(demoScore.getInfo());
```

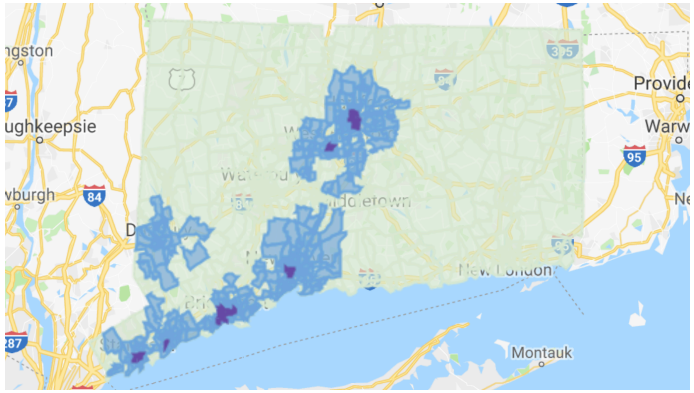


Figure: Top 50 tracts with High Demographic Score

```
var demo_sorted = demoScore.sort('demoScore',false);
var demo_high = demo_sorted.limit(50);
Map.addLayer(demo_high, {color:'674EA7'}, 'Top 50 Tracts with Higher Demo Score');
```

Link to code: https://code.earthengine.google.com/628e45d937abf443f560e38f5500aa52

Step three. Compute the Economic Score.

```
//Economic Score;
//create a function to compute income level and house value level;

function econ_score(feature)
{
  var income = ee.Number(feature.get('Med_Inc')).log();
  var house = ee.Number(feature.get('Med_Value')).log();
  var incomeScore = 0;
  //low wage line in Connecticut is $28,720.
  if (income < 28720 & house < 150000)
  {
    incomeScore = 100;
  }
  //median wage in Connecticut is $74,168.
  if (income >= 28720 & income < 74168)
  {
    incomeScore = 40;
  }
  else
  {
    incomeScore = 20;
  }

  return feature.set({"incomeScore":incomeScore});
}

var econScore = ee.FeatureCollection(demoScore.map(econ_score));
print(econScore.getInfo());



var econ_sorted = econScore.sort('incomeScore',false);
var econ_high = econ_sorted.limit(50);
Map.addLayer(econ_high, {color:'F1C232'}, 'Top 50 Tracts with Higher Economic Score');
```

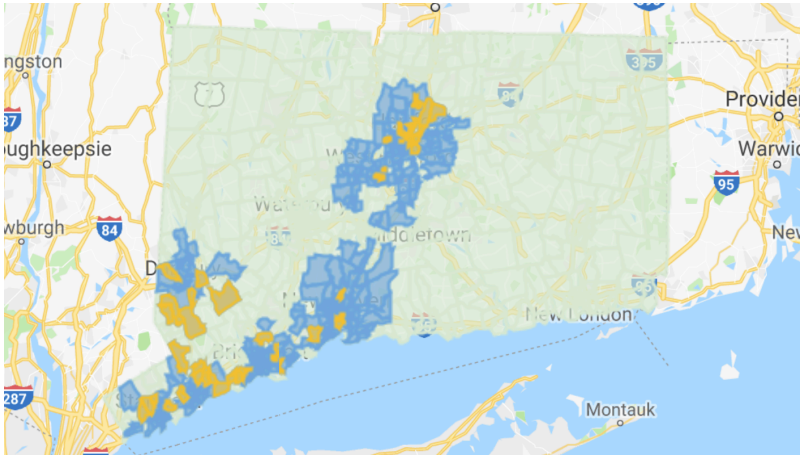


Figure: Top 50 tracts with High Economic Score

Link to code: https://code.earthengine.google.com/628e45d937abf443f560e38f5500aa52

Step four. Compute the Social Score.

We assign weights to social factors: 0.3*Pct_AfAm + 0.4*Pct_BlPov + 0.3*Pct_MF_DU

```
//Social Score;
//Black Americans tend to support gun control;
//Population below poverty level tend to support gun control;
//People living in multiple family is more likely to support gun control;


function social_score(feature)
{
  var AfricanAmerican_Sc = ee.Number(feature.get('Pct_AfAm')).multiply(0.3);
  var BelowPoverty_Sc = ee.Number(feature.get('Pct_BlPov')).multiply(0.4);
  var MultiFamily_Sc = ee.Number(feature.get('Pct_MF_DU')).multiply(0.3);
  var social_Sc = AfricanAmerican_Sc.add(BelowPoverty_Sc).add(MultiFamily_Sc);
  return feature.set({"social_Sc":social_Sc});
}
var socialScore = ee.FeatureCollection(econScore.map(social_score));
print(socialScore.getInfo());

var social_sorted = socialScore.sort('social_Sc',false);
var social_high = social_sorted.limit(50);
Map.addLayer(social_high, {color:'E06666'}, 'Top 50 Tracts with Higher Social Score');
```

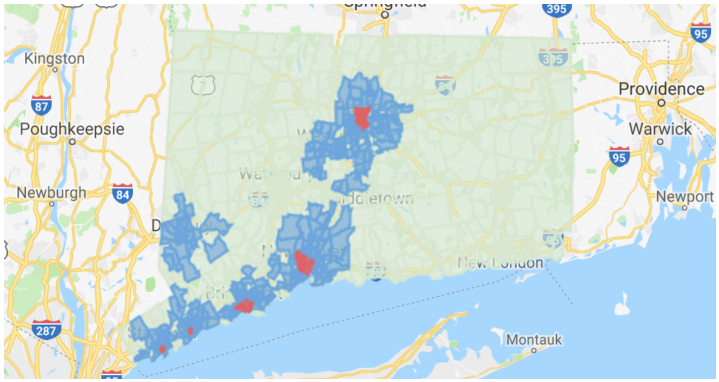


Figure: Top 50 tracts with High Social Score.

Link to code: https://code.earthengine.google.com/628e45d937abf443f560e38f5500aa52

Step five: Compute Final Score.

```
//Compute Final Score;
function final_score(feature)
{
  var demosc = ee.Number(feature.get('demoScore')).multiply(50);
  var econsc = ee.Number(feature.get('incomeScore')).multiply(30);
  var socialsc = ee.Number(feature.get('social_Sc')).multiply(20);
  var finalsc = demosc.add(econsc).add(socialsc);
  return feature.set({"finalsc":finalsc});
}

var finalScore = ee.FeatureCollection(socialScore.map(final_score));
print(finalScore.getInfo());

var finalSc_sort = finalScore.sort('finalsc',false);
print(finalSc_sort);
var support_guncontrol50 = finalSc_sort.limit(50);
//Map.addLayer(support_guncontrol50, {color:'45818E'}, 'Top 50 Tracts Most Likely to Support Gun Control');
var support_guncontrol10 = finalSc_sort.limit(10);
//Map.addLayer(support_guncontrol10, {color:'4C1130'}, 'Top 10 Tracts Most Likely to Support Gun Control');
var support_guncontrol5 = finalSc_sort.limit(5);
Map.addLayer(support_guncontrol5, {color:'0C343D'}, 'Top 5 Tracts Most Likely to Support Gun Control');
```

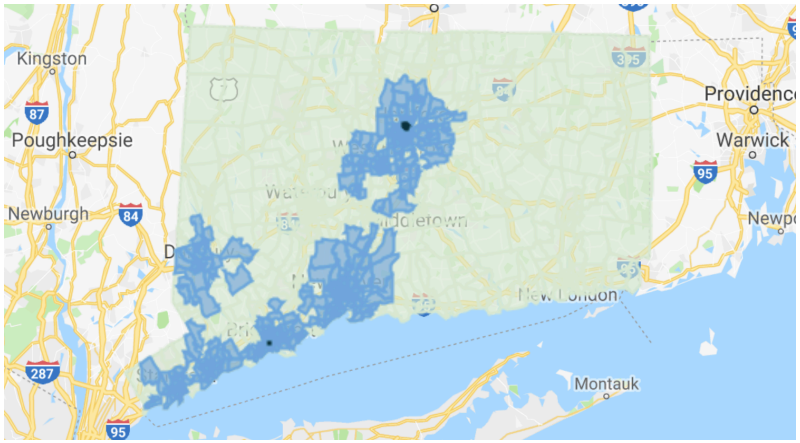


Figure: Top 5 Tracts Most Likely to Support Gun Control

Link to code: https://code.earthengine.google.com/628e45d937abf443f560e38f5500aa52

Assignment 03

We want to get a closer look at the five tracts:

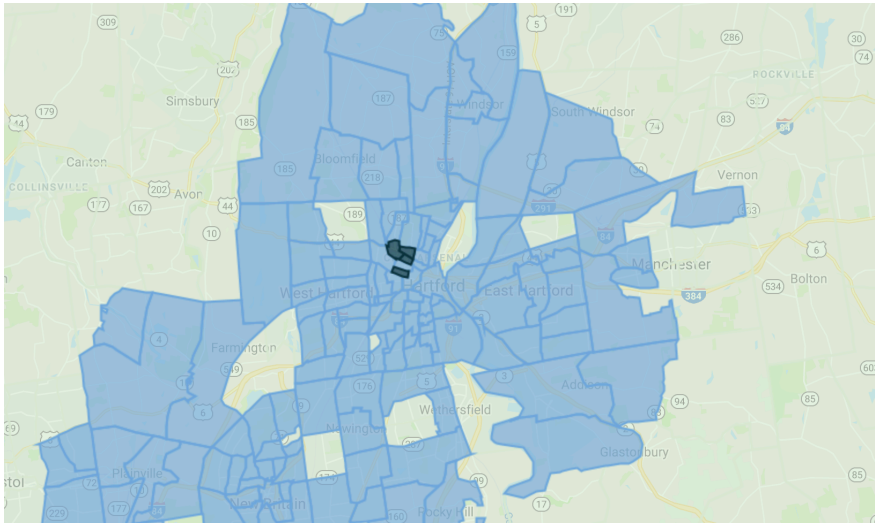
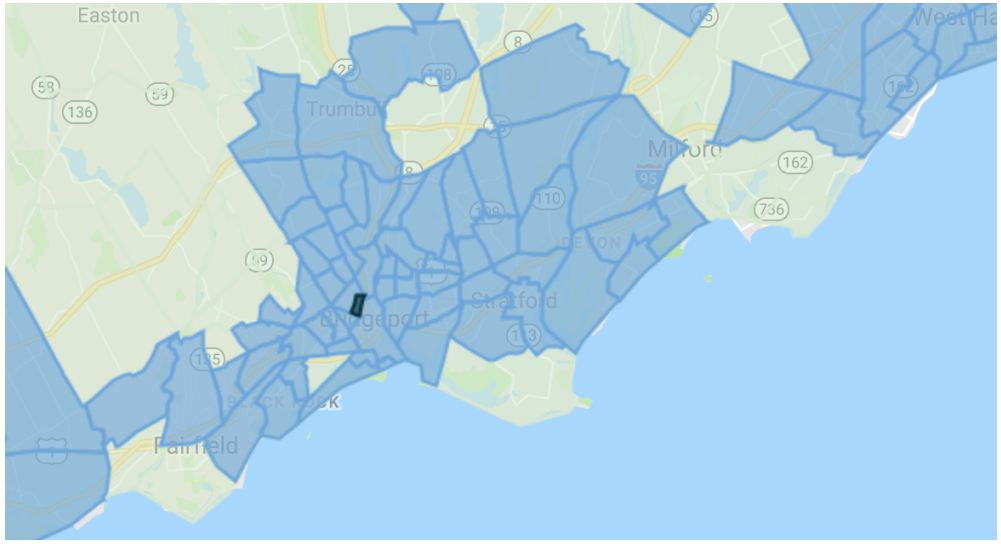


Figure: Four tracts in Hartfold. One tract in Bridgeport.



Link to code: https://code.earthengine.google.com/628e45d937abf443f560e38f5500aa52