Final Project Report

Air Pollution and Respiratory Disease Issue in South Korea

| Meen Wook Jung |

Community and Regional Planning

- **Introduction**

South Korea is recently experiencing an unprecedented air pollution problem. An increase not only in pollutants coming from outside but also in the usage of domestic fossil fuel has been combined with climate change, resulting in a pollutant stagnation in South Korea's atmosphere.

As a result, the level of air quality was marked as the worst ever on March 5th, 2019. Moreover, Greenpeace's analysis of the 2019 World Air Quality Report, the overall air quality in South Korea was recorded as the worst level among OECD member countries. Considering that the pollutants are composed of toxic ingredients including ozone, carbon monoxide, and sulfur dioxide, they must be harmful to the human body, especially to the respiratory system.

Even though the air quality is the primary concern to many citizens in South Korea, there is no enough data on how a micro-sized particle called fine dust had affected mortality. In order to address this issue, it is necessary to investigate the connection between fine dust and respiratory system damage. Therefore, this project aims not only to visualize fine dust and mortality data but also to analyze their correlation.

- **Objectives**

(1) Identify the fine dust concentrations and respiratory diseases-related mortality in South Korea.

(2) Examine the correlation between the pollutant concentrations and mortality.

(3) Create interactive maps to visualize (1) and (2)

- **Web Mapping Tools and Technologies**

- *Tools*
  - GitHub: a repository of mapping data and HTML code
  - QGIS: a GIS software for converting .shp format to .geojson.
- *Technologies*
  - Leaflet: a Javascript library for making interactive maps
  - Leaflet Plugin: plugins for displaying various data formats in a leaflet map
  - GeoJSON: an open standard data format for geographic features
  - Choropleth Map: a colorful interactive map for visualizing data
  - Layer Control Button: the creation of a control button for switching the form of maps
  - Legend: an explanation of a choropleth map
  - Pop-up Window: a window box for presenting data attributes

- **Data Source**

– The level of fine dust air pollution

https://sgis.kostat.go.kr/view/thematicMap/thematicMapMain?stat_thema_map_id=9pyrpJvwHw20160121115806991GvpLyuuwDt&theme=CTGR_005&mapType=05&CTGRS=CTGR_001:recommend,CTGR_002:recommend,CTGR_003:recommend,CTGR_004:recommend,CTGR_005:recommend

– The record of respiratory disease mortality

http://kosis.kr/statHtml/statHtml.do?orgId=101&tblId=DT_1B34E13&conn_path=I2

- **Methodologies**

1. *Leaflet and Leaflet Plugin*

```
    <link rel="stylesheet" href="https://unpkg.com/leaflet@1.6.0/dist/leaflet.css" integrity="sha512-
xwE/Az9zrjBIphAcBb3F6JVqxf46+CDLwfLMHIoNu6KEQCAWi6HcDUbeOfBIptF7tcCzusKFjFw2yuvEpDL9wQ==" crossorigin="" />
    <script src="https://unpkg.com/leaflet@1.6.0/dist/leaflet.js" integrity="sha512-
gZwIG9x3wUXg2hdXF6+rVkLF/0Vi9U8D2Ntg4Ga5I5BZpVkVxIJWbSQtXPSiUTtC0TjtGOmxa1AJPuV0CPthew==" crossorigin=""></script>
```

```
      <div id='map2'></div>

         <script type="text/javascript" src="https://andyjung82.github.io/CRP558/final_project_geojson/pm/pm.geojson"></script>

            <script type="text/javascript">

              var map2 = L.map('map2').setView([35.65664, 127.81270], 7);

              L.tileLayer('https://api.mapbox.com/styles/v1/{id}/tiles/{z}/{x}/{y}?
access_token=pk.eyJ1IjoibWFwYm94IiwiYSI6ImNpejY4NXVycTA2emYycXBndHRqcmZ3N3gifQ.rJcFIG214AriISLbB6B5aw', {
                    maxZoom: 18,
                    attribution: 'Map data &copy; <a href="https://www.openstreetmap.org/">OpenStreetMap</a> contributors, ' +
                        '<a href="https://creativecommons.org/licenses/by-sa/2.0/">CC-BY-SA</a>, ' +
                        'Imagery © <a href="https://www.mapbox.com/">Mapbox</a>',
                    id: 'mapbox/light-v9',
                    tileSize: 512,
                    zoomOffset: -1
                }).addTo(map2);
```

After building a basic HTML layout, I inserted leaflet plugin script for creaing an interactive map in the contents. Because I created two interactive maps, I distinguished the two maps by using a different division such as <div id='map1'> and <div id='map2'>.

2. *GeoJSON*

```
<script type="text/javascript" src="https://andyjung82.github.io/CRP558/final_project_geojson/pm/pm.geojson"></script>
```

```
<script type="text/javascript" src="https://andyjung82.github.io/CRP558/final_project_geojson/mor/mor.geojson"></script>
```

```
var pm14 = new L.geoJson(pm, {
    style: mystyle6,
    onEachFeature: onEachFeature6
}).addTo(map2);

var pm15 = new L.geoJson(pm, {
    style: mystyle7,
    onEachFeature: onEachFeature7
}).addTo(map2);

var pm16 = new L.geoJson(pm, {
    style: mystyle8,
    onEachFeature: onEachFeature8
}).addTo(map2);

var pm17 = new L.geoJson(pm, {
    style: mystyle9,
    onEachFeature: onEachFeature9
}).addTo(map2);

var pm18 = new L.geoJson(pm, {
    style: mystyle10,
    onEachFeature: onEachFeature10
}).addTo(map2);
```

```
var mor14 = new L.geoJson(mor, {
    style: mystyle1,
    onEachFeature: onEachFeature1
}).addTo(map);

var mor15 = new L.geoJson(mor, {
    style: mystyle2,
    onEachFeature: onEachFeature2
}).addTo(map);

var mor16 = new L.geoJson(mor, {
    style: mystyle3,
    onEachFeature: onEachFeature3
}).addTo(map);

var mor17 = new L.geoJson(mor, {
    style: mystyle4,
    onEachFeature: onEachFeature4
}).addTo(map);

var mor18 = new L.geoJson(mor, {
    style: mystyle5,
    onEachFeature: onEachFeature5
}).addTo(map);
```

After collecting and pre-processing data I planned to use, I applied to the first and second scripts for setting up the geojson files' link. Next, I assigned the geojson files in the Javascript variable. The geojson file named "pm" contains the fine dust data from 2014 to 2018 and "mor" is composed of Pneumonia and Chronic Obstructive Pulmonary Disease-related mortality in the same period.

3. *Choropleth Map*

```
function color(d) {
    return d > 60.600001 ?'#ab4124' :
        d > 54.500001 ? '#bf5f3f' :
        d > 50.250001 ? '#cf7d5d' :
        d > 47.000001 ? '#e0a082' :
        d > 45.500001 ? '#edc3a8' :
        '#fae9d4';
}

function mystyle6(feature) {
    return {
        weight: 0.5,
        opacity: 1,
        color: 'black',
        fillOpacity: 1,
        fillColor: color(feature.properties.PM10_14)
    };
}
```

```
function color(d) {
    return d > 194.600001 ?'#000000' :
        d > 167.500001 ? '#333333' :
        d > 125.250001 ? '#666666' :
        d > 85.000001 ? '#999999' :
        d > 60.500001 ? '#cccccc' :
        '#ffffff';
}

function mystyle1(feature) {
    return {
        weight: 0.5,
        opacity: 1,
        color: 'black',
        fillOpacity: 1,
        fillColor: color(feature.properties.MOR_14)
    };
}
```

Because I used two different types of data, it was necessary to use different ranges of intervals. With the aid of the natural break (Jenks) method, I divided the intervals into six.

4. *Legend*

```
<style>
    .legend {
        line-height: 10px;
        color: #555;
    }
    .legend i {
        width: 18px;
        height: 18px;
        float: left;
        margin-right: 8px;
        opacity: 0.9;
    }
    .info {
        padding: 6px 8px;
        font: 10px/20px Arial, Helvetica, sans-serif;
        background: rgba(255,255,255,0.8);
        box-shadow: 0 0 15px rgba(0,0,0,0.2);
        border-radius: 5px;
        width: 150px;
    }
    .info h4 {
        margin: 0 0 5px;
        color: #777;
    }
</style>
```

```
var legend2 = L.control({
      position: 'bottomright'
  });

    legend2.onAdd = function(map) {

        var div = L.DomUtil.create('div', 'info legend'),
            grades = [30, 45, 47, 50, 54, 60],
            labels = ["<b>Particulate Matter 10 <br> <i>(μm)</i></b>"],
            from, to;

        for (var i = 0; i < grades.length; i++) {
            from = grades[i];
            to = grades[i + 1];

            labels.push(
                '<i style="background:' + color(from + 1) + '"></i> ' +
                from + (to ? '&ndash;' + to : '-70'));
        }

        div.innerHTML = labels.join('<br>');
        return div;
    };

    legend2.addTo(map2);
```

After applying to the choropleth colors, I made legends for showing the ranges on the maps. The first box shows the CSS attribute of the legend and the second box presents the necessary script for inserting a legend on an interactive map.

5. Layer Control Button

```
      var baseMaps2 = {
          "Particulate matter 2014": pm14,
          "Particulate matter 2015": pm15,
          "Particulate matter 2016": pm16,
          "Particulate matter 2017": pm17,
          "Particulate matter 2018": pm18,
        };

L.control.layers(baseMaps2).addTo(map2);
```

Since my data was composed of time-series, there were lots of variables. So I decided to visualize the annual data attribute through the layer control button. As shown in the box, the assigned Javascript variables (pm14, pm15, pm16, pm17, and pm18) were assigned to each layer.

6. Interactive Graph

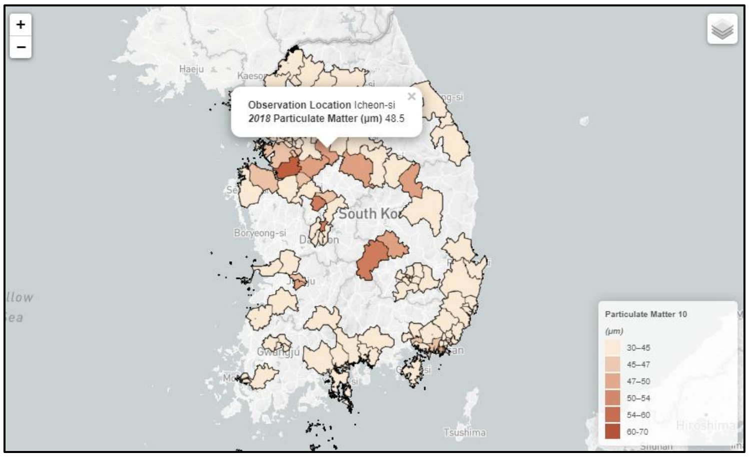
```
<script src="https://d3js.org/d3.v4.js"></script>
```

```
d3.csv("https://raw.githubusercontent.com/andyjung82/CRP558/master/final_project_csv/data_re.csv", function(data) {

var x = d3.scaleLinear()
  .domain([25, 70])
  .range([ 0, width ]);
svg.append("g")
  .attr("transform", "translate(0," + height + ")")
  .call(d3.axisBottom(x));

var y = d3.scaleLinear()
  .domain([0, 350])
  .range([ height, 0]);
svg.append("g")
  .call(d3.axisLeft(y));
```

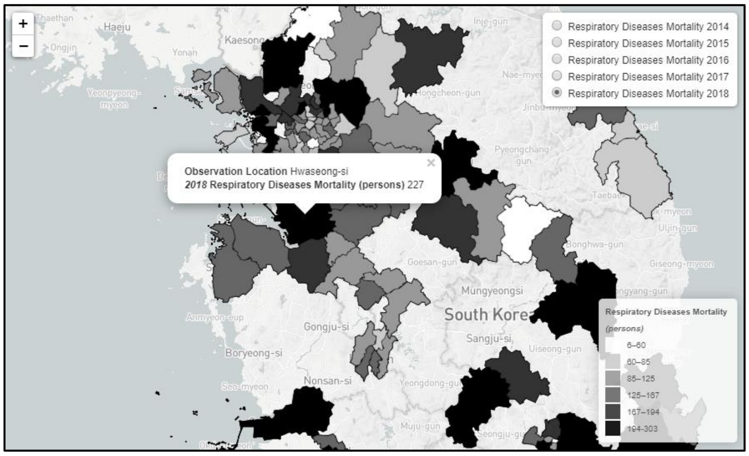
```
          var mouseover = function(d) {
            tooltip
              .style("opacity", 1)
          }

          var mousemove = function(d) {
            tooltip
              .html(d.SIGUNGU + "<br> <i>2018 <b>Particulate Matter:</b></i> " + d.PM10_18 + " <i>(µm)</i> <br> <i>2018 <b>Respiratory-related Mortality:
</b></i> " + d.MOR_18 + " <i>(persons)</i>")
              .style("left", (d3.mouse(this)[0]+90) + "px") // It is important to put the +90: other wise the tooltip is exactly where the point is an it
creates a weird effect
              .style("top", (d3.mouse(this)[1]) + "px")
          }

          var mouseleave = function(d) {
            tooltip
              .transition()
              .duration(200)
              .style("opacity", 0)
          }
```

My original intention was to display a scatter plot in pop-up windows. However, because I used time-series data, it was hard to display all the data in a map. So, I utilized the interactive graph instead of it.

7. *Results*

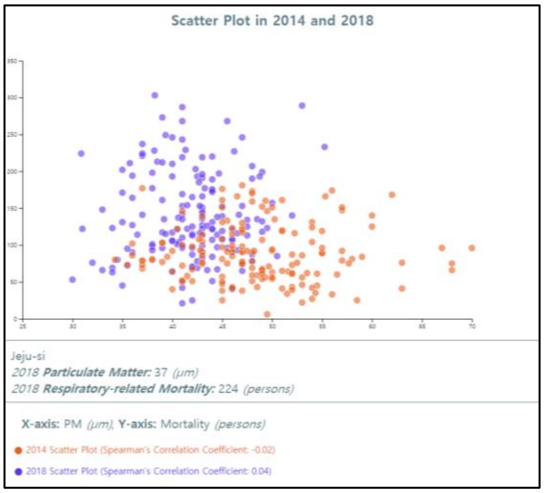


Observation Location Icheon-si
2018 **Particulate Matter (μm)** 48.5



Observation Location Hwaseong-si
2018 **Respiratory Diseases Mortality (persons)** 227

○ Respiratory Diseases Mortality 2014
○ Respiratory Diseases Mortality 2015
○ Respiratory Diseases Mortality 2016
○ Respiratory Diseases Mortality 2017
● Respiratory Diseases Mortality 2018



Scatter Plot in 2014 and 2018

Jeju-si
2018 ***Particulate Matter:*** 37 *(μm)*
2018 ***Respiratory-related Mortality:*** 224 *(persons)*

X-axis: PM *(μm)*, **Y-axis:** Mortality *(persons)*

● 2014 Scatter Plot (Spearman's Correlation Coefficient: -0.02)

● 2018 Scatter Plot (Spearman's Correlation Coefficient: 0.04)

- **Evaluation**

I realized that using interactive maps for visualizing how fine dust was distributed is an effective way. This is because the distribution of fine dust and mortality is a phenomenon that occurs based on location. Even though I failed to get a meaningful correlation between the two variables, I think that continuous data building and visualization are necessary for public health in the future.