Spatial Analysis of Toronto City

Introduction

In the following analysis, I have opted for various Machine Learning techniques to analyze the geographic typology and the socio-economic environment of Toronto city. I have done this by analyzing a dataset that houses different information about Toronto. To do this efficiently I have also downloaded two external datasets - one having the location of different retirement homes, and the other having the location of different skill development agencies for youths. This has helped me to reach the conclusion stated below

In [ ]:
```
import geopandas # used to extend the datatypes used by pandas to allow
 spatial operations on geometric types
import pandas #pandas is mainly used for data analysis
import contextily as ctx #contextily is used to add a basemap underneat
h the data points.
from pysal.viz import mapclassify #mapclassify implements a family of c
lassification schemes for choropleth maps. It determines the number of
 classes, and the assignment of observations to those classes
from pysal.lib import weights #used to calculate the spatial weights ma
trix.
import seaborn as sns #Seaborn is a Python data visualization library b
ased on matplotlib. It provides a high-level interface for drawing attr
active and informative statistical graphics
import matplotlib.pyplot as plt #Matplotlib is a plotting library for t
he Python programming language and its numerical mathematics extension
 NumPy.
from sklearn import cluster #The sklearn library contains a lot of effi
```

```
cient tools for machine learning and statistical modeling including cla
ssification, regression, clustering and dimensionality reduction
```

In [ ]:
```
#read geopackage file
#neis is an object in which the data of the geopackage files are stored
neis = geopandas.read_file("https://darribas.org/gds_course/_downloads/
a2bdb4c2a088e602c3bd6490ab1d26fa/toronto_socio-economic.gpkg")
```

In [2]:
```
# Dataset for supporting second variable
#ab is the object in which the data of the geopackage files are stored
ab=geopandas.read_file("C:/Users/Souvik/Desktop/Geographic Data Scienc
e/Retirement Homes.gpkg")
```

In [3]:
```
# Dataset for supporting first variable
#ab is the object in which the data of the geopackage files are stored
br=geopandas.read_file("C:/Users/Souvik/Desktop/Geographic Data Scienc
e/Employment Resources.gpkg")
```

In [4]:
```
#db is the object in which the data of the comma-seperated values file
 is stored
db=pandas.read_csv("https://darribas.org/gds_course/_downloads/8944151f
1b7df7b1f38b79b7a73eb2d0/toronto_socio-economic_vars.csv")
```

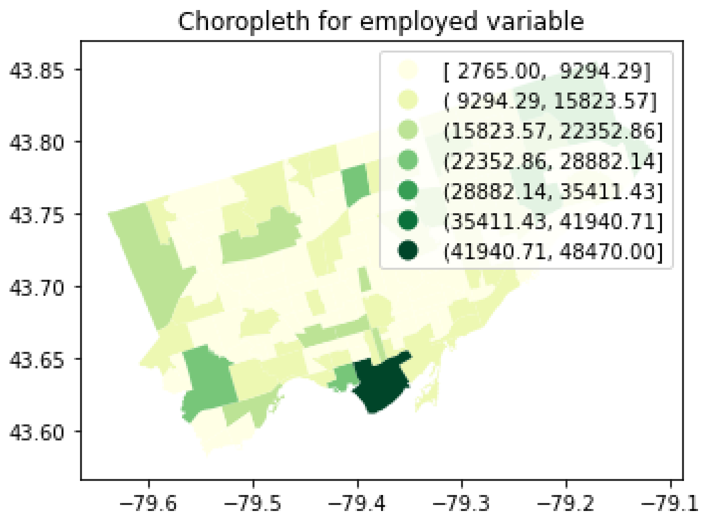
In [5]:
```
#The coordinate referencing system or crs is used to represnt three-dim
esional object on a two dimensional plane.
#To transform data into latitudes and longitudes we use EPSG code 4326.
neis.crs = "EPSG:4326"
```

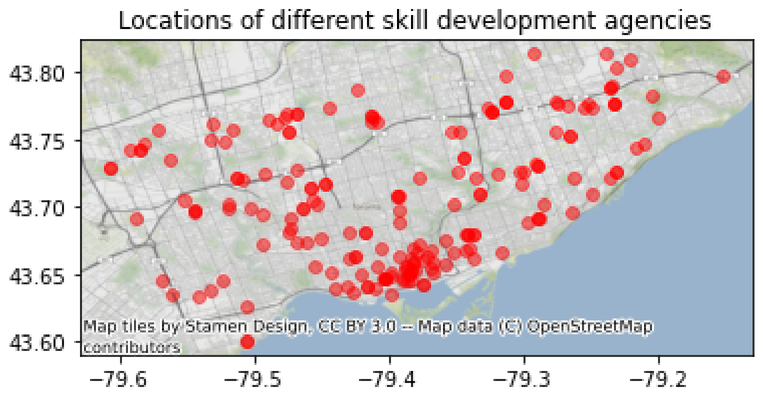
Part I

In the below "choropleth for employed variable" figure, we get different shades of green where the darkest portion shows the least number of employed people and the lighter portions show the most number of people who are employed. I have chosen another dataset from the Toronto Open Data Portal. This is the dataset that shows youth development programs in the different regions of Toronto. We can observe that most youth development programs from the "location of different skill development agencies" are centered in the darkest regions' of the choropleth hence proving that employment is low in those regions

In [6]:
```
#Inspiration for designing choropleths(https://medium.com/using-special
ist-business-databases/creating-a-choropleth-map-using-geopandas-and-fi
nancial-data-c76419258746)
ax=neis.plot(column="employed", #choosing the first variable employed a
nd finding out its choropleths
         scheme="equalinterval", #choosing the equalinterval scheme whi
ch partitions the bins in such an order that the number of counts are i
n a decreasing order
         k=7, #choosing the number of bins to be 7
         cmap="YlGn", #Choosing the colour green from the Color Brewer
         legend="True" #legend is set to True so that the different bin
s are showed in the map.

       )
ax.set_title("Choropleth for employed variable")#setting the title of t
he map


# Plotting the polygons
ax = br.plot(alpha=0.5, color='red'); #alpha value depicts the opaquene
ss of the dots. Red is the color chosen for the dots
ctx.add_basemap(ax, crs=br.crs) # adding basemap
ax.set_title("Locations of different skill development agencies")#setti
ng the title for the basemap induced map
```

Out[6]:
```
Text(0.5, 1.0, 'Locations of different skill development agencies')
```





Locations of different skill development agencies
Map tiles by Stamen Design, CC BY 3.0 -- Map data (C) OpenStreetMap contributors
43.80
43.75
43.70
43.65
43.60
−79.6
−79.5
−79.4
−79.3
−79.2

In [8]:
```
# Creating the spatial weights matrix
%time w = weights.Queen.from_dataframe(neis, idVariable= "_id")
#spatial weights matrix is made using the _id column, as the spatial we
ights matrix just shows the neighbours of polygons not neighbours of th
e specific variable.
```

```
Wall time: 479 ms
```

In [9]:
```
# Row standardize the matrix
w.transform = 'R'
```

To make the operations easier we make a standardized version of the variable which we are working with. This can be done by subtracting the average value and divide by the standard deviation of each observation of the column.

In [11]:
```
neis['employed_std'] = (neis['employed'] - neis['employed'].mean()) / n
eis['employed'].std()
```

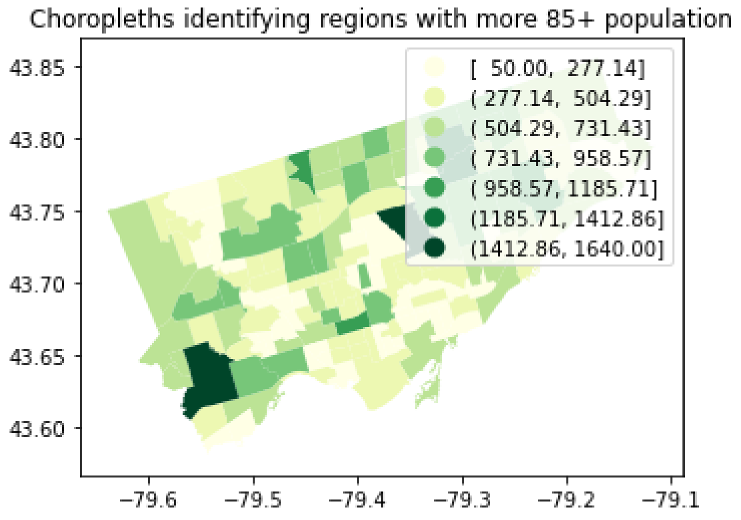
In [12]:
```
#We create the spatial lag of the variable
neis['w_employed_std'] = weights.lag_spatial(w, neis['employed_std'])
```

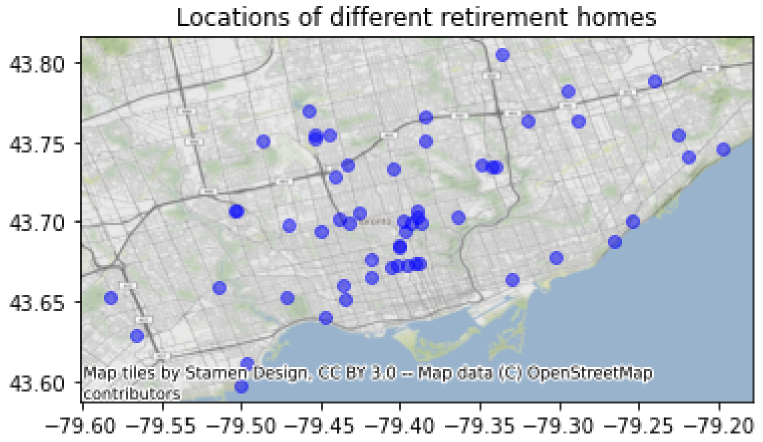
In the below "Choropleths identifying regions with more 85+ population" figure we get different shades of green where the darkest portion shows the least number of people with age 85+ and the lighter portions show more people with 85+ population. I have chosen another dataset from the Toronto Open Data Portal. This is the dataset that shows the locations of different retirement homes in the different regions of Toronto. We can observe that most retirement homes from the "location of different retirement homes" are centered in the lighter regions' of the choropleth hence proving that the number of people with age 85+ is more in those regions

In [41]:
```
ax=neis.plot(column="pop_85+_yearsold", #choosing the second variable p
op_85+_yearsold and finding out its choropleths
        scheme="equalinterval", #choosing the equalinterval scheme whic
h partitions the bins in such an order that the number of counts are in
 a decreasing order
         k=7, #choosing the number of bins to be 7
        cmap="YlGn", #Choosing the colour green from the Color Brewer
        legend="True" #legend is set to True so that the different bins
 are showed in the map.
       )

ax.set_title("Choropleths identifying regions with more 85+ population"
) #setting the title of the first  map
```

```
import contextily as ctx #contextily is used to add a basemap underneat
h the data points.
# Plot polygons
ax = ab.plot(alpha=0.5, color='blue'); #alpha denotes the opaqueness of
 the dots. The color of the dots are blue
# Adding background map, expressing target CRS so the basemap can be

ctx.add_basemap(ax, crs=ab.crs) # adding basemap
ax.set_title("Locations of different retirement homes") #setting the ti
tle of the second map
```

```
Out[41]: Text(0.5, 1.0, 'Locations of different retirement homes')
```




Locations of different retirement homes
43.80
43.75
43.70
43.65
43.60
−79.60 −79.55 −79.50 −79.45 −79.40 −79.35 −79.30 −79.25 −79.20
Map tiles by Stamen Design, CC BY 3.0 -- Map data (C) OpenStreetMap contributors

To make the operations easier we make a standardized version of the variable which we are working with. This can be done by subtracting the average value and divide by the standard deviation of each observation of the column

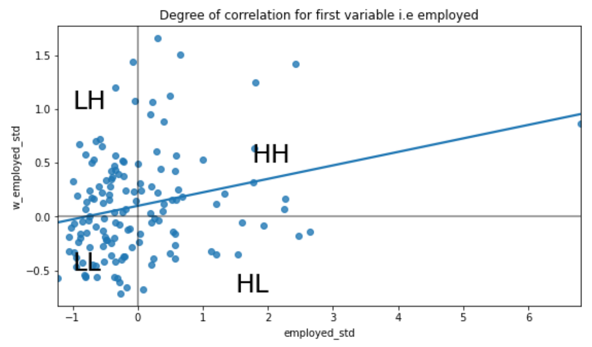
In [16]:
```
neis['pop_85+_std'] = (neis['pop_85+_yearsold'] - neis['pop_85+_yearsol
d'].mean()) / neis['pop_85+_yearsold'].std()
```

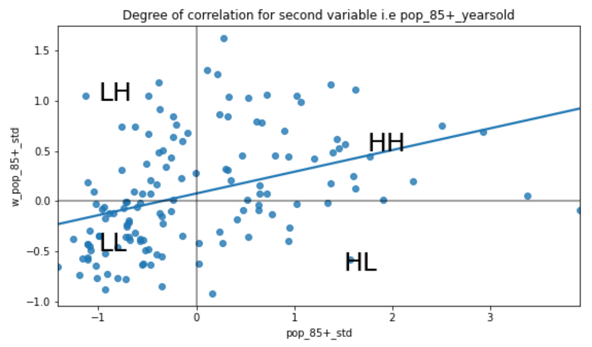
In [17]:
```
#We create the spatial lag of the variable
neis['w_pop_85+_std'] = weights.lag_spatial(w, neis['pop_85+_std'])
```

In [19]:
```
f, axs = plt.subplots(1,2, figsize=(20, 5))#we have one row of pictures
 and two colomns that represent 1,2. The size of the figures are 20 X 5
# Plot values
sns.regplot(x='employed_std', y='w_employed_std', data=neis, ci=None,ax
=axs[0]) #x axis holds 'employed_std' and y axis holds 'w_employed_std'
 and the plot is made of the employed variable from the data stored in
 neis object

# Add vertical and horizontal lines
```

```
axs[0].axvline(0, c='k', alpha=0.5)# axvline() function from pyplot mod
ule is used to add vertical lines across the axes of the plot
axs[0].axhline(0, c='k', alpha=0.5)# axhline() funtion from pyplot modu
le is used to add horizontsasl lines across the axes of the plot
axs[0].text(1.75, 0.5, "HH", fontsize=25) #Setting the Location of text
 'HH' in plot and fontsize 25
axs[0].text(1.5, -0.7, "HL", fontsize=25) #Setting the locationg of tex
t 'HL' in plot and fontsize 25
axs[0].text(-1, 1, "LH", fontsize=25) #Setting the location of text 'L
H' in plot and fontsize 25
axs[0].text(-1, -0.5, "LL", fontsize=25) #Settign the location of text
 'LL' in plot and fontsize 25
axs[0].set_title("Degree of correlation for first variable i.e employe
d")#settign the title of first figure from left
# Display
plt.show()

#Inspiration for bringing images side by side(https://stackoverflow.co
m/questions/17079279/how-is-axis-indexed-in-numpys-array)


# Plot values

sns.regplot(x='pop_85+_std', y='w_pop_85+_std', data=neis, ci=None, ax=
axs[1]) #x asix holds 'pop_85+_std' and y axis holds 'w_pop_85+_std' an
d the plot is made of the employed variable from the data stored in nei
s object
# Add vertical and horizontal lines
axs[1].axvline(0, c='k', alpha=0.5)# axvline() function from pyplot mod
ule is used to add vertical lines across the axes of the plot
axs[1].axhline(0, c='k', alpha=0.5)# axhline() funtion from pyplot modu
le is used to add horizontsasl lines across the axes of the plot
axs[1].text(1.75, 0.5, "HH", fontsize=25)#Setting the Location of text
 'HH' in plot and fontsize 25
axs[1].text(1.5, -0.7, "HL", fontsize=25)#Setting the locationg of text
 'HL' in plot and fontsize 25
axs[1].text(-1, 1, "LH", fontsize=25)#Setting the location of text 'LH'
 in plot and fontsize 25
axs[1].text(-1, -0.5, "LL", fontsize=25)#Settign the location of text
```

```
 'LL' in plot and fontsize 25
axs[1].set_title("Degree of correlation for second variable i.e pop_85+
_yearsold")#settign the title of first figure from left
# Display
plt.show()
```





The first figure from the left named "Degree of correlation for first variable i.e. employed" shows a positive correlation. But the data points are denser in the low -low region indicating that similar low values are together hence indicating more number of low employment regions. Also, the number of data points in the low-high region is more indicating clusters of dissimilar values or regions with a mixed degree of employment. There are also data points with high high regions meaning high employment areas. But they are denser in the low low region but not evenly spread out in the High high region indicating more low employment regions.

The second figure from the left named "Degree of correlation of second variable i.e. pop_85+_yearsold" also shows a positive correlation. The data here is almost evenly spread out in the low- low and high- high region. Meaning there is a region where the population of 85+ years of people is high and also regions where a population of 85+ is low.

PART-II

In [33]:
```
#These are the parameters which I have chosen to indentify the differen
t kinds of neighborhoods
```

```
parameters=['bedrooms_4+','bedrooms_3','employed','pop_25-54_yearsold',
'deg_medics']
```

I have chosen the above mentioned parameters to indentify this type of typology as these will give me an idea of the living standard of the people living in Toronto. Hence this will help me to identify the regions which are poverty stricken.

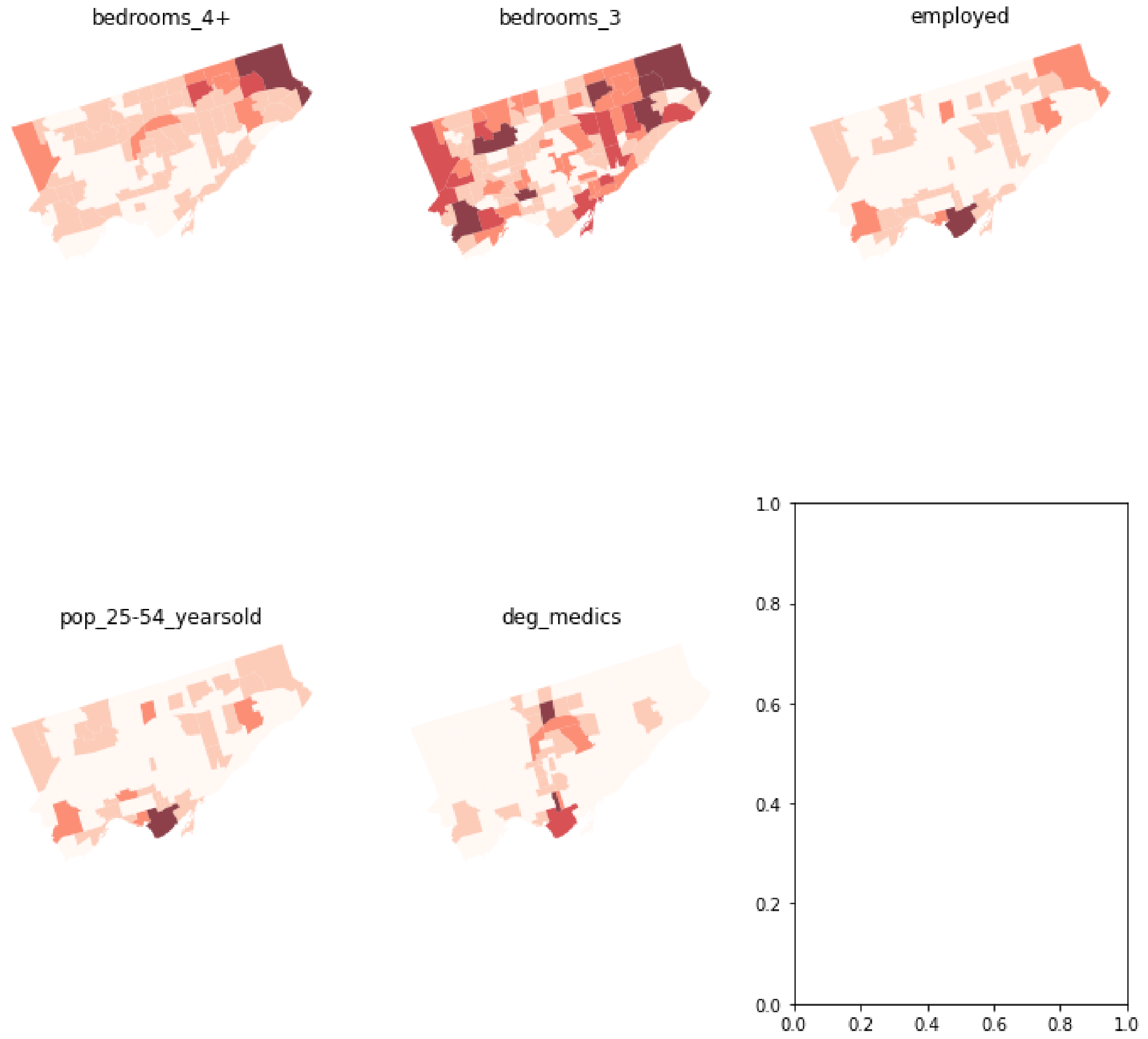
In [34]:
```
kmeans5 = cluster.KMeans(n_clusters=5, random_state=12345) #Using k mea
ns clusterig technique to make 5 clusters. Random state is used to make
 a random slection of data points as k means selects centroids from ran
dom region and finds the euclidean distance in them.
k5cls = kmeans5.fit(neis[parameters])#parameters hard-coded above is se
t to fit in those clusters
neis['k5cls'] = k5cls.labels_ #all the combination of different paramet
ers are assigned different numbers. Each number represents a different
 category, so two observations with the same number belong to same grou
p
```

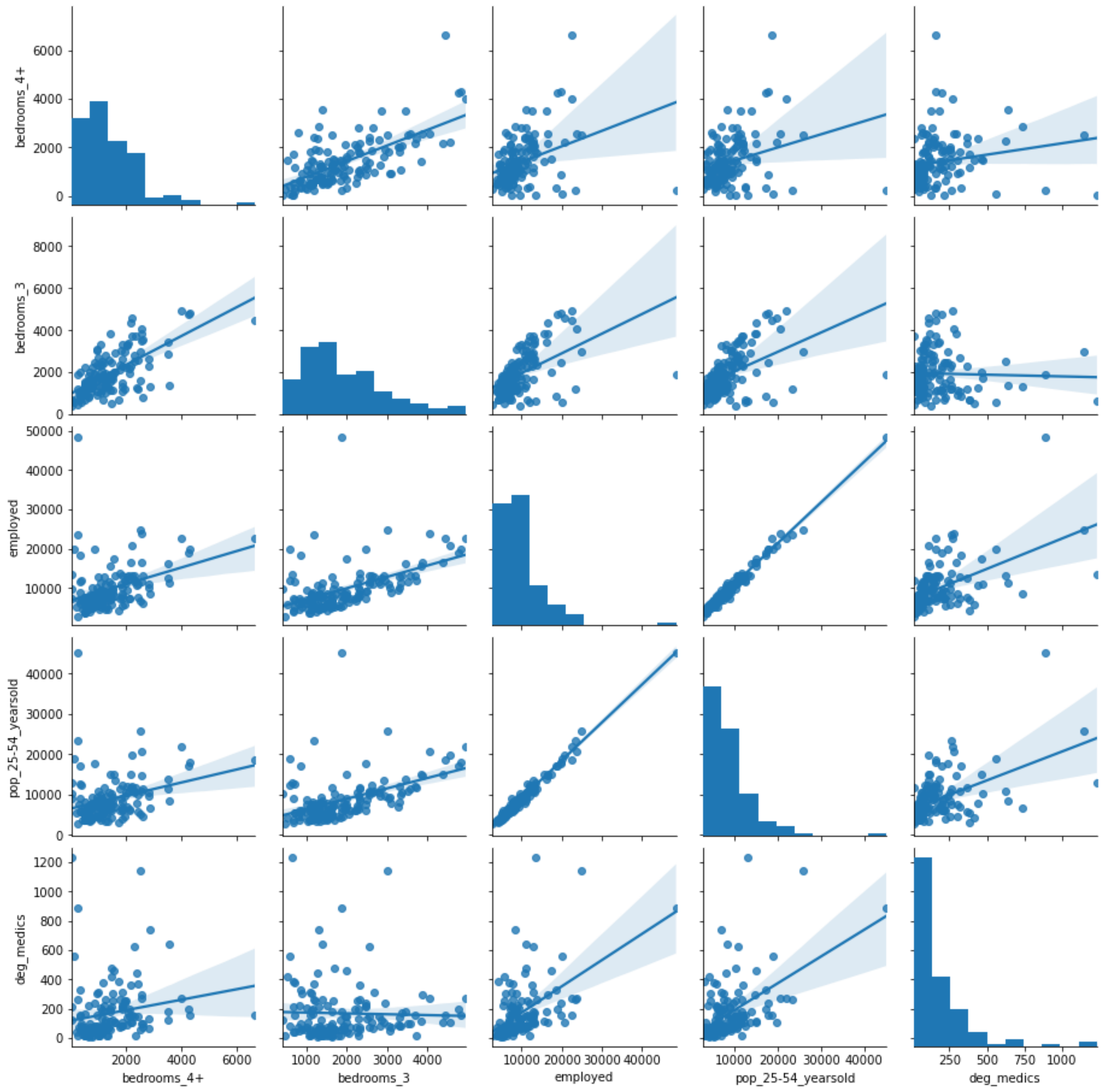
</p>We can figure out from the visual representations of the parameters that the wealthiest areas of the city are located in the middle portion and the eastern portion of Toronto City. I have reached this conclusion by plotting each of the above-mentioned parameters into the map of Toronto. The darker regions show the dominance of these parameters. We can also notice that there are people with medical degrees coming up from the least employed region of the city. So if we can provide more support to that region the country will flourish much faster as it has no shortage of talented people</p>

In the next histogram I have plotted each variable in the parameter list with every other variable. We can see that there is a positive coorelation with the employed variable and the bedrooms_4+, bedrooms_3 and medical degree holder. Hence with regions of high employment there are more number of localities close by having higher number of bedrooms and higher number of people receiving better education

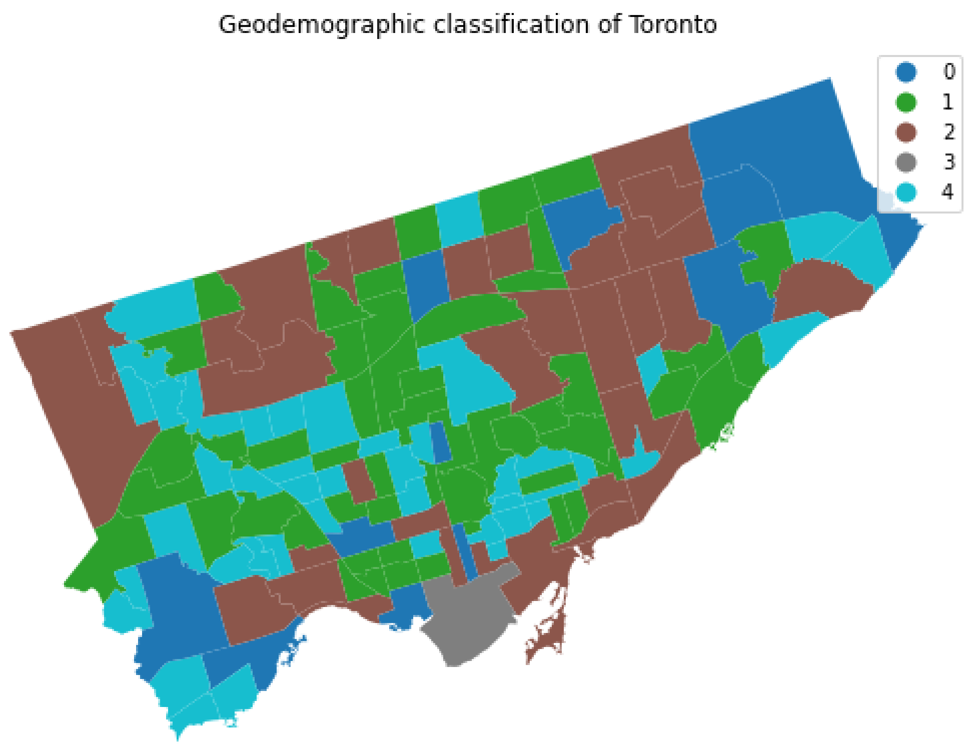
I have made 5 clusters of the variables in the paramater list. The last map shows the areas with these clusters

In [43]:
```
# Create figure and axes (this time it's 6, arranged 2 by 3)
f, axs = plt.subplots(nrows=2, ncols=3, figsize=(12, 12))#No of rows fo
r first histogram is 2 and columns is 3. The size of the figures are 12
 X 12
# Make the axes accessible with single indexing
axs = axs.flatten()
# Start the loop over all the variables of interest
for i, col in enumerate(parameters):
    # select the axis where the map will go
    ax = axs[i]
    # Plot the map
    neis.plot(column=col, ax=ax, scheme='EqualInterval', \
             linewidth=0, cmap='Reds', alpha=0.75)
    # Remove axis clutter
    ax.set_axis_off()
    # Set the axis title to the name of variable being plotted
    ax.set_title(col)
# Display the figure
plt.show()

_ = sns.pairplot(neis[parameters], kind='reg', diag_kind='hist')


# Setup figure and ax
f, ax = plt.subplots(1, figsize=(9, 9))
# Plot unique values choropleth including a legend and with no boundary
 lines
neis.plot(column='k5cls', categorical=True, legend=True, linewidth=0, a
x=ax)
# Remove axis
ax.set_axis_off()
# Add title
plt.title('Geodemographic classification of Toronto')
# Display the map
plt.show()
```


bedrooms_4+
bedrooms_3
employed
pop_25-54_yearsold
deg_medics
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0



bedrooms_4+
bedrooms_3
employed
pop_25-54_yearsold
deg_medics



The table below shows the different clusters and the mean values of each variable. So if one wants to settle in an area of high employment he or she might look out for areas falling in the fourth cluster

```
In [29]: # Calculate the mean by group
```

```
k5means = neis.groupby('k5cls')[parameters].mean()
# Show the table transposed
k5means.T
```

Out[29]:

k5cls	0	1	2	3	4
bedrooms_4+	2545.454545	1280.384615	1855.15625	230.0	905.340909
bedrooms_3	3243.181818	1756.057692	2725.78125	1870.0	1246.931818
employed	21362.727273	8357.788462	12831.09375	48470.0	5290.340909
pop_25-54_yearsold	19882.272727	7531.153846	11710.31250	45105.0	4535.340909
deg_medics	336.818182	158.750000	220.31250	890.0	83.522727

Conclusion

After analyzing the Toronto Data set using various state-of-the-art algorithms and techniques I have concluded that though there is a good rate of employment in this city some parts of the city are very much neglected. But they don't lack in any kind of talented people. So if these areas can be taken care of then the country will fourish more. Also, I felt that maybe increasing the number of retirement homes in areas having more people above the 85+ age will lead to a much better healthcare system.

Bibliography

Location of retirement homes

Locations of agencies' providing youth development programs

Inspiration for bringing images side by side

"Inspiration for designing choropleths "

"More in depth understanding of K-means clustering technique"

In [ ]:

```
```