

Preparation to run the code

Na Jiang

Before running the code, you may need to check the following points.

1. Need Python 3 installed
 - Mac: <https://docs.python-guide.org/starting/install3/osx/>
 - Windows: <https://www.youtube.com/watch?v=UvcQIPZ8ecA>
2. Jupyter Notebook or Jupyter Lab are recommended to run the code.
 - Jupyter Notebook: <https://test-jupyter.readthedocs.io/en/latest/install.html>
 - Jupyter Lab: https://jupyterlab.readthedocs.io/en/stable/getting_started/installation.html
3. If necessary, Install the packages that are needed for the generation process, which are: **networkx, geopandas, pandas, random, sklearn, pickle.**
4. Change the data path (i.e., red box shown in Figure 1) to where you store these data.



```
1, Read Data

1.1, Road Network
In: road = gpd.read_file('Data/Road_line.shp')

1.2, Demographic data (Census Tract)
In: dp = gpd.read_file('Data/Census_Cleaned.shp').set_index('GEOID10')
    #check size of census tract data
    len(dp)

1.3, Commute Information
In: od = pd.read_csv('Data/tract-od.csv.zip', dtype={i:(str if i<2 else int) for i in range(6)})
In: od.head()

1.4, Workplace
In: cbp = pd.read_csv('Data/cbp10co.zip')
    cbp = cbp[(cbp.naics_str.startswith('-'))] #All types of establishments included
    cbp['fips'] = cbp.fipstate.map("{:02}".format) + cbp.fipscty.map("{:03}".format)
    cbp = cbp.set_index('fips')
```

Figure 1, Change Path

5. If you are willing to generate the synthetic population other than our NYC area, you may want to download the data that are needed for your study area and operate the preprocess by yourselves. In addition, step 0 code only provides what we have done for the preprocessing, there may be more or less works based on various study area.
6. Figure 2 shows the cell that applies all function to generate the synthetic population. You may need to change the results file name in the red box based on your needs.

4, Apply above functions to generate Population

```
In [ ]: #list to hold population
population = []
#hold error
errors = []
#hold workplaces info
wps = []

#Test Code
data = dp[0:2]
data.apply(lambda t: synthesize(t,od,road,errors, population, wps),axis=1);

#Main Apply, Whole Study Area, apply those funtion on each line of the processed cenese tract data
#dp.apply(lambda t: synthesize(t,od,road,errors, population, wps),axis=1);

#Hold the results in pickles for later ectraction, which will save memory
with open('Results/'+errors+"3000".pkl, 'wb') as f:
    pickle.dump(errors, f)
with open('Results/'+population+"3000".pkl, 'wb') as f:
    pickle.dump(population, f)
with open('Results/'+wps+"3000".pkl, 'wb') as f:
    pickle.dump(wps, f)
```

Figure 2, Results path

7. To replicate the rest of steps (i.e., Step 2,3,4), you may need you save the results from code “1_Creat_Individuals” into “.csv” file. In code “1_Creat_Individuals”, section 4.1 shows how to convert the data to “.csv” file.
8. Step 2 and 3 can be intergraded into the code “1_Creat_Individuals”, because these steps are optional based on different purpose of project. In our case, we need to assign the Road ID to each agent in order to run our model, where we use Road ID to place agents on the map. If you are not intended to do so, you don’t need Step 3.
9. If you have any other questions related to the code, you can contact me at njiang8@gmu.edu. I will try my best to get back to you within 24h.