



SI 507 Lab #4



September 21/22



Schedule

Lab #4

Lab resources: [REMOVED]

- Logistics
 - HW2
 - Schelling's model
 - Context
 - Examples
 - Homework code
 - Overview
 - Intuition
 - Tips + Worktime
-

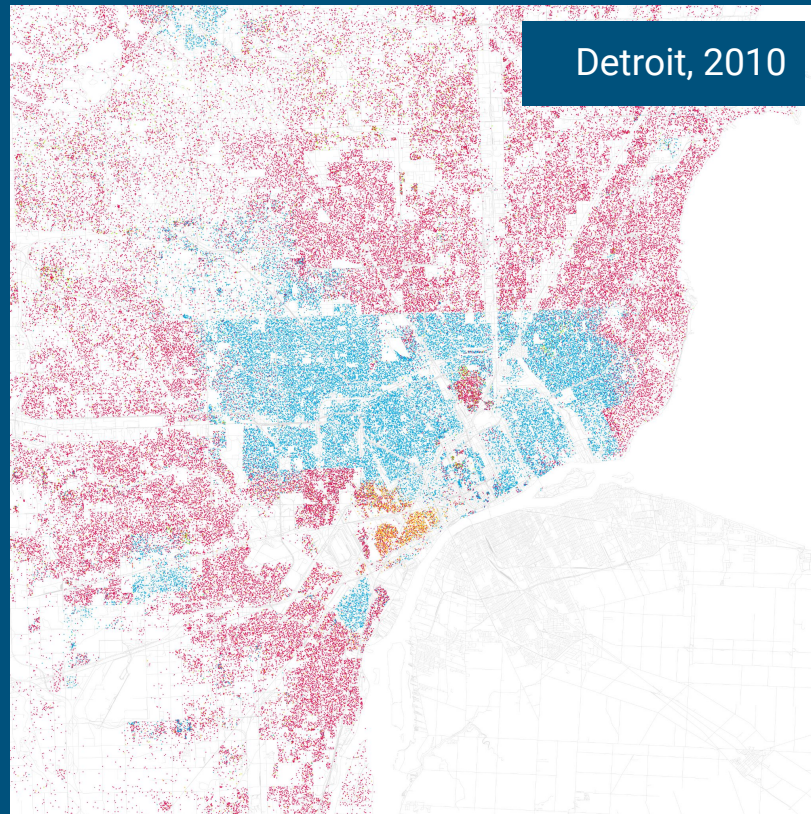
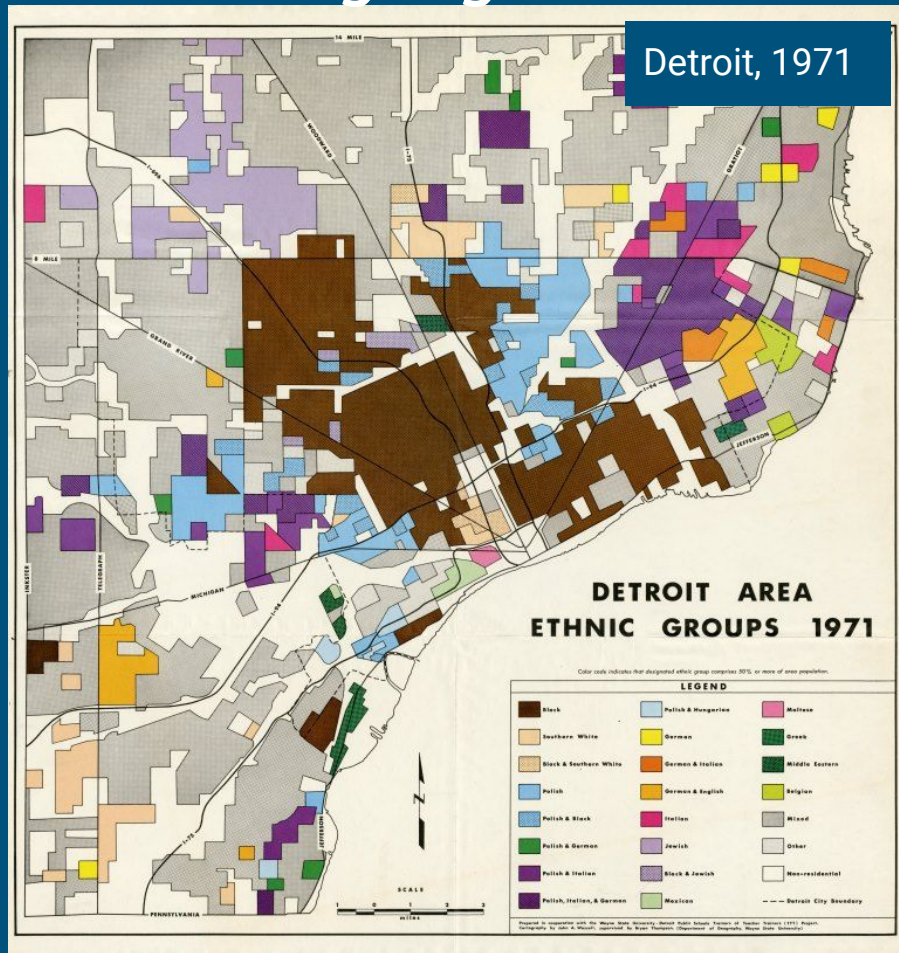
Logistics

Lab #4

- [Removed]



Racial Segregation in the United States



"Red is White, Blue is Black, Green is Asian, Orange is Hispanic, Gray is Other, and each dot is 25 people."

Racial Segregation in the United States

- Trying to explain racial segregation: **Schelling's model** ([link](#))
 - "mechanisms that translate unorganized individual behavior into collective results." (Schelling, 1971)
 - Agent-based modeling
- Context (1970's)
 - Race riots
 - Redlining
 - Racially restrictive covenants
- → Only believe Schelling if you think that racial segregation can be fully explained by individual behavior based on free choice of where to live

Schelling's model of segregation (Example 1)

The algorithm ($t=0.5$, $n=4$):

- Mark all unhappy households
 - A household is unhappy if its group (O/+) makes up less than t of the neighborhood
 - A household's neighborhood includes itself and its n closest neighbors
- For each unhappy household (left \rightarrow right):
 - if the household is still unhappy, then move it to the closest place to be not unhappy
 - otherwise, don't move it
- Repeat until no unhappy households

Example:

O+000++O+00++00 [Start]

???????????????? [Iteration 1]

Schelling's model of segregation (Example 1)

The algorithm ($\tau=0.5$, $n=4$):

- Mark all unhappy households
 - A household is unhappy if its group (O/+) makes up less than τ of the neighborhood
 - A household's neighborhood includes itself and its n closest neighbors
- For each unhappy household (left \rightarrow right):
 - if the household is still unhappy, then move it to the closest place to be not unhappy
 - otherwise, don't move it
- Repeat until no unhappy households

Example:

O+000++0+00++00 [Start]

0000++++++00000 [Iteration 1]

Schelling's model of segregation (Example 2)

The algorithm ($t=0.5$, $n=4$):

- Mark all unhappy households
 - A household is unhappy if its group (O/+) makes up less than t of the neighborhood
 - A household's neighborhood includes itself and its n closest neighbors
- For each unhappy household (left \rightarrow right):
 - if the household is still unhappy, then move it to the closest place to be not unhappy
 - otherwise, don't move it
- Repeat until no unhappy households

Example:

00++00++00++0+0 [Start]

???????????????? [Iteration 1]

Schelling's model of segregation (Example 2)

The algorithm ($t=0.5$, $n=4$):

- Mark all unhappy households
 - A household is unhappy if its group (O/+) makes up less than t of the neighborhood
 - A household's neighborhood includes itself and its n closest neighbors
- For each unhappy household (left \rightarrow right):
 - if the household is still unhappy, then move it to the closest place to be not unhappy
 - otherwise, don't move it
- Repeat until no unhappy households

Example:

00++00++00++0+0 [Start]

0000++++++000+0 [Iteration 1]

0000++++++0000 [Iteration 2]

Schelling's model of segregation (Example 2)

The algorithm ($t=0.5$, $n=5$):

- Mark all unhappy households
 - A household is unhappy if its group (O/+) makes up less than t of the neighborhood
 - A household's neighborhood includes itself and its n closest neighbors
- For each unhappy household (left \rightarrow right):
 - if the household is still unhappy, then move it to the closest place to be not unhappy
 - otherwise, don't move it
- Repeat until no unhappy households

Example:

00++00++00++0+0 [Start]

0000++++++000+0 [Iteration 1]

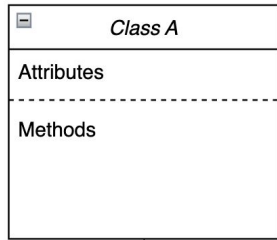
0000++++++0000 [Iteration 2]

\rightarrow HW2: Each household is an "Agent"

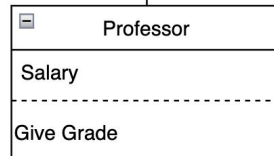
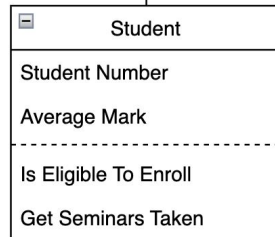
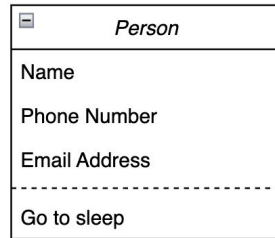
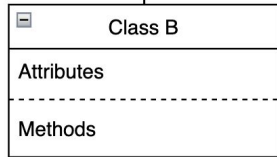
HW2 (Overview)

- Overview
 - Implement a simplified version of the Schelling model
 - Use Python classes and inheritance to model Agents (i.e. households)
 - Suggested due [DUE DATE]
- Package requirements
 - `IPython`
 - `matplotlib`
- Parts
 - Lab notebook
 - HW2 code
- What's going on in the code???
 - Let's sketch it out...

Object Oriented Diagram (Examples)



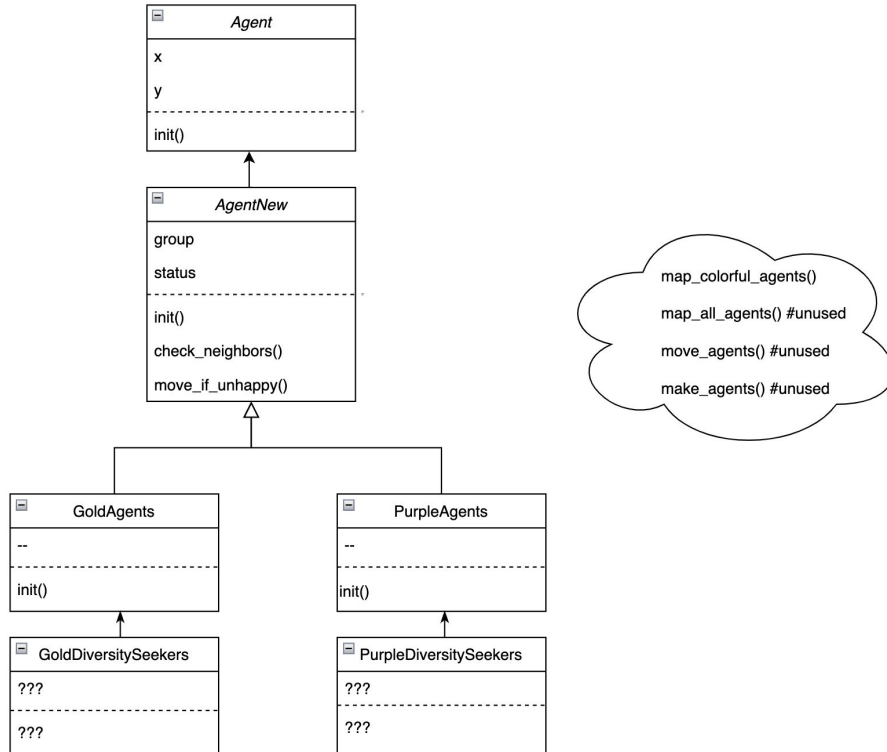
Inherits



Takeaways:

- A class is made out of:
 - attributes
 - methods
- Child class inherits attributes and methods from parent class
- Overriding methods is OK

Object Oriented Diagram (HW2)



HW2 (Workflow)

- Workflow 1
 - Copy HW2 to a notebook (each problem in a separate cell)
 - Work through all of lab_4 in order
 - Work through all of hw_2 in order
- Workflow 2
 - Copy HW2 to a notebook (each problem in a separate cell)
 - Work through an order that makes sense in the diagram