MAKING SPATIAL ACCESS METRICS MORE ACCESSIBLE: Documentation for the PySAL Spatial Access Package

Package URL: https://access.readthedocs.io/
Package Development: James Saxon
Documentation: Karina Acosta Ordonez and Julia Koschinsky
Research Assistance: Dan Snow, Vidal Anguiano, Yair Atlas, Bryan Wang
What is in this documentation?

• This presentation overviews and compares the catchment area and rational agent access models implemented in the Spatial Access for PySAL package

• Section 1: The Package + Ecosystem

• Section 2: Overview, definitions and data inputs of each model

• Section 3: Equations, interpretations, and assumptions of each model

• Section 4: Visual and numerical examples of each model
SECTION 1: THE PACKAGE + ECOSYSTEM
New Spatial Access Package
https://access.readthedocs.io/

Spatial Access for PySAL

Whether you work with data in health, retail, employment or other domains, spatial accessibility measures help identify potential spatial mismatches between the supply and demand of services. They indicate how close demand locations are to supply locations.

Motivation

We built this package for several reasons:
- to make the new spatial access metric (RAAM) available,
- to allow for easy comparison between RAAM and classic spatial access models,
- to support spatial access research at scale by making pre-computed travel time matrices available and sharing code for computing new matrices at scale, and
- to allow users who prefer a point-and-click interface to obtain spatial access results for their data using our web app (for US).

Methods

This PySAL package implements our new measure that simultaneously accounts for travel time and congestion at the destination:

- Rational Agent Access Model (RAAM) [Saxon and Snow 2019, SS19].

Here is an example of the results of the RAAM model from this article: It shows how spatially accessible each Census tract is to primary care, compared to the national average. Darker blue areas have better spatial access (below-average travel costs) while darker red areas have worse spatial access (above average travel costs).
# API reference

## Accessibility Class

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>access.access</code> (demand_df, demand_value, ...)</td>
<td>Spatial Access Class</td>
</tr>
<tr>
<td><code>access.access.raam</code> (self, name, cost, ...)</td>
<td>Calculate the rational agent access model.</td>
</tr>
<tr>
<td><code>access.access.fca_ratio</code> (self, name, ...)</td>
<td>Calculate the floating catchment area (buffer) ratio access score.</td>
</tr>
<tr>
<td><code>access.access.two_stage_fca</code> (self, name, ...)</td>
<td>Calculate the two-stage floating catchment area access score.</td>
</tr>
<tr>
<td><code>access.access.three_stage_fca</code> (self, name, ...)</td>
<td>Calculate the three-stage floating catchment area access score.</td>
</tr>
<tr>
<td><code>access.access.score</code> (self, col_dict, name)</td>
<td>Weighted aggregate of multiple already-calculated, normalized access components.</td>
</tr>
<tr>
<td><code>access.access.euclidean_distance</code> (self, ...)</td>
<td>Calculate the Euclidean distance from demand to supply locations.</td>
</tr>
<tr>
<td><code>access.access.euclidean_distance_neighbors</code> (self)</td>
<td>Calculate the Euclidean distance among demand locations.</td>
</tr>
<tr>
<td><code>access.access.user_cost</code> (self, new_cost_df, ...)</td>
<td>Create a user cost, from demand to supply locations.</td>
</tr>
<tr>
<td><code>access.access.user_cost_neighbors</code> (self, ...)</td>
<td>Create a user cost, from supply locations to other supply locations.</td>
</tr>
</tbody>
</table>

## Access Functions

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>access.raam.raam</code> (demand_df, supply_df, cost_df)</td>
<td>Calculate the rational agent access model's total cost.</td>
</tr>
<tr>
<td><code>access.fca.weighted_catchment</code> (loc_df, cost_df)</td>
<td>Calculation of the floating catchment (buffered) accessibility sum, from DataFames with computed distances.</td>
</tr>
<tr>
<td><code>access.fca.fca_ratio</code> (demand_df, supply_df, ...)</td>
<td>Calculation of the floating catchment accessibility</td>
</tr>
<tr>
<td><code>access.fca.two_stage_fca</code> (demand_df, ...)</td>
<td>Calculation of the floating catchment accessibility</td>
</tr>
<tr>
<td><code>access.fca.three_stage_fca</code> (demand_df, ...)</td>
<td>Calculation of the floating catchment accessibility</td>
</tr>
</tbody>
</table>

- ✔️ Rational Agent Access Model (RAAM)
- ✔️ Floating Catchment Areas (FCA)
- ✔️ 2-Stage FCAs (2SFCA)
- ✔️ Enhanced 2SFCA (E2SFCA)
- ✔️ 3-Stage FCA (3SFCA)
- ✔️ Access Score
Workflow for Package

Data Input

Add Origin + Destination Data
demand + supply values

Add Externally Computed Travel Matrices
Euclidean distance on the fly

Travel Times

Spatial Access Metrics

PySAL Package:
RAAM
Floating Catchment Models
Access Score

Data Output

Results added to Origins
<table>
<thead>
<tr>
<th>Name</th>
<th>Installation</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>pgRouting</td>
<td>docker</td>
<td>Good for driving, open-source and free, PostgreSQL/postgis and OpenStreetMap (OSM)</td>
</tr>
<tr>
<td>OSRM</td>
<td>install / R / docker</td>
<td>Best for driving, OSM, open-source and free, customized travel parameters, C++</td>
</tr>
<tr>
<td>Open Trip Planner</td>
<td>docker routing / resources / DockerHub</td>
<td>Best for transit, open-source and free, customized travel parameters, Java</td>
</tr>
<tr>
<td>Valhalla Open Source Routing</td>
<td>install</td>
<td>Multi-modal, OSM, open-source, for fee at scale, Python</td>
</tr>
<tr>
<td>Pandana</td>
<td>install</td>
<td>Good for driving and walking, OSM, open-source and free, part of UrbanSim, Python</td>
</tr>
<tr>
<td>Graphhopper</td>
<td>install</td>
<td>Multi-modal, OSM, open-source, for fee at scale, Python</td>
</tr>
<tr>
<td>Spatial Access Package</td>
<td>install / notebooks</td>
<td>Best for walking, OSM, scales well, open-source and free, includes spatial access metrics, Python</td>
</tr>
<tr>
<td>Google Maps</td>
<td>install</td>
<td>Accurate multi-modal, customized travel parameters, commercial, expensive at scale</td>
</tr>
</tbody>
</table>

Tools for Computing Your Own Travel Time Matrices

https://access.readthedocs.io/en/latest/resources.html

https://pypi.org/project/spatial-access/
Pre-Computed Travel Time Matrices

https://access.readthedocs.io/en/latest/resources.html

- **3 Modes:** Walking, Transit + Driving
- **National:** Tract-Level
- **20 Largest Cities:** Tracts + Blocks
- **Generated with OpenTrip Planner**

<table>
<thead>
<tr>
<th>CITY NAME</th>
<th>COUNTY GEOID</th>
<th>Walking</th>
<th>Transit</th>
<th>Driving</th>
</tr>
</thead>
<tbody>
<tr>
<td>National</td>
<td>All</td>
<td>Tracts</td>
<td>Tracts</td>
<td>Tracts</td>
</tr>
<tr>
<td>New York City (Manhattan)</td>
<td>36061</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>New York City (Brooklyn)</td>
<td>36047</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>New York City (The Bronx)</td>
<td>36005</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>New York City (Staten Island)</td>
<td>36085</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>New York City (Queens)</td>
<td>36061</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>Los Angeles</td>
<td>06037</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>Chicago</td>
<td>17031</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks (Part 1)</td>
<td>Tracts / Blocks (Part 2)</td>
</tr>
<tr>
<td>Houston</td>
<td>48201</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>Phoenix</td>
<td>04013</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>Philadelphia</td>
<td>42101</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>San Antonio</td>
<td>48029</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>San Diego</td>
<td>06073</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>Dallas</td>
<td>48113</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>San Jose</td>
<td>06085</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>Austin</td>
<td>48453</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>Jacksonville</td>
<td>12031</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>Fort Worth</td>
<td>48439</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>Columbus</td>
<td>39049</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>San Francisco</td>
<td>06075</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>Charlotte</td>
<td>37119</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>Indianapolis</td>
<td>18097</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>Seattle</td>
<td>53033</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>Denver</td>
<td>08031</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
<tr>
<td>Washington D.C.</td>
<td>11001</td>
<td>Tracts / Blocks</td>
<td>Tracts / Blocks</td>
<td>Tracts</td>
</tr>
</tbody>
</table>

Created by: Dan Snow, MPP | Sept. 2019
Create the access object.

Since we have geometries here, it could be as simple as this:

```python
In [43]: A = access(demand_df = il_med(["pop", "geometry"]), demand_value = 'pop',
              supply_df = il_med(["doc", "dentist", "geometry"]), supply_value = ["doc", "dentist"],
              demand_index = True, supply_index = True,
              cost_df = times, cost_name = 'cost',
              cost_origin = 'origin', cost_dest = 'dest',
              neighbor_cost_df = times, neighbor_cost_name = 'cost',
              neighbor_cost_origin = 'dest', neighbor_cost_dest = 'origin')
```

In this case, we would have to generate the distance matrix on the fly.

Instead, I'll create a slightly more complicated version with all of the possible matrices:

```python
In [44]: A = access(demand_df = il_med(["pop", "geometry"]), demand_value = 'pop',
              supply_df = il_med(["doc", "dentist", "geometry"]), supply_value = ["doc", "dentist"],
              demand_index = True, supply_index = True,
              cost_df = times, cost_name = 'cost',
              cost_origin = 'origin', cost_dest = 'dest',
              neighbor_cost_df = times, neighbor_cost_name = 'cost',
              neighbor_cost_origin = 'dest', neighbor_cost_dest = 'origin')
```

```python
In [47]: A.weighted_catchment  (name = "gravity", weight_fn = gravity)
A.fca_ratio  (name = "fca", max_cost = 15)
A.fca_ratio  (name = "fca", max_cost = 30) # Note - the warning -- good!
A.fca_ratio  (name = "fca90", max_cost = 60)
A.fca_ratio  (name = "fca90", max_cost = 90)
A.two_stage_fca  (name = "2sfca", max_cost = 60)
A.enhanced_two_stage_fca(name = "2sfca30", weight_fn = fn30)
A.enhanced_two_stage_fca(name = "2sfca60", weight_fn = fn60)
A.enhanced_two_stage_fca(name = "q2sfca", weight_fn = gaussian)
A.three_stage_fca  (name = "3sfca")
A.rram(name = "rram", tau = 60);  
A.rram(name = "rram30", tau = 30);
```

Results time!

Let's get the correlations among measures, and plot a few of them...

```python
In [51]: A.norm_access_df.columns
```
SECTION 2: OVERVIEW, DEFINITIONS AND DATA INPUTS
## DIMENSIONS OF ACCESS

<table>
<thead>
<tr>
<th>Spatial</th>
<th>Potential</th>
<th>Realized</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Potential use of nearby services – assumed in PySAL spatial access package</td>
<td>Actual use of nearby services</td>
</tr>
<tr>
<td>Aspatial</td>
<td>Aspatial determinants of <em>latent</em> service demand, such as income, age or sex</td>
<td>Aspatial drivers of <em>observed</em> service demand such as insurance or language</td>
</tr>
</tbody>
</table>
Floating Catchment Areas (FCA): For each provider, this is the ratio of providers to clients within a given travel time to the provider (Huff 1963, Joseph and Bantock 1982, and Luo 2004).

Two-Stage FCAs (2SFCA): Calculated in two steps for a given travel time to the provider: 1) for each provider, the provider-to-client ratio is generated, 2) for each point of origin, these ratios are then summed (Luo and Wang, 2002, and Wang and Luo 2005).

Enhanced 2SFCA (E2SFCA): 2SFCA but with less weight to providers that are still within the travel threshold but at larger distances from the point of origin (Luo and Qi 2009).


Access Score: This is a weighted sum of access components like distance to provider and relative importance of provider type (Isard 1960).

Rational Agent Access Model (RAAM): A model that optimizes the allocation of patients to providers by minimizing travel times and congestion at the provider (Saxon and Snow 2019).
References

## ACCESS MODEL TIMELINES AND ACRONYMS

<table>
<thead>
<tr>
<th>Year</th>
<th>Acronym</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>1960</td>
<td>ACS</td>
<td>Access Score</td>
</tr>
<tr>
<td>1963</td>
<td>FCA</td>
<td>Standard Floating Catchment Area</td>
</tr>
<tr>
<td>2002</td>
<td>2SFCA</td>
<td>Two-stage Floating Catchment Areas</td>
</tr>
<tr>
<td>2009</td>
<td>E2SFCA</td>
<td>Enhanced two-stage Floating Catchment Areas</td>
</tr>
<tr>
<td>2012</td>
<td>3SFCA</td>
<td>Three-stage Floating Catchment Areas</td>
</tr>
<tr>
<td>2019</td>
<td>RAAM</td>
<td>Rational Agent Access Model</td>
</tr>
</tbody>
</table>
## MODEL COMPARISON

<table>
<thead>
<tr>
<th>FCA</th>
<th>2SFCA</th>
<th>E2SFCA</th>
<th>3SFCA</th>
<th>RAAM</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
</tr>
<tr>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
</tr>
<tr>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
</tr>
<tr>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
</tr>
<tr>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
<td><img src="icons/checkmark.png" alt="Checkmark" /></td>
</tr>
</tbody>
</table>

- **PTD ratio for providers**
  - Ratio provider to demand (population)

- **... aggregated at origin with fixed distance**
  - Unweighted sum of ratios within catchment area without

- **Selection weight**
  - Uses a step function to weight distance within catchment

- **Impedance weight**
  - Takes into account the probability of selection based on distance to provider

- **Service congestion**
  - Cost of competition
• **Catchment area**: area within which provider is *accessible* to people (varies between models)
• **Travel time**: commuting time or distance covered by residents to reach suppliers’ location (in RAAM models, this is called travel cost)
• **Congestion cost**: competing demand on a provider (only used in RAAM)

Main inputs of spatial potential access models

1. **Location of providers/supply** (e.g. XY coordinates or Census tract geo ID)
2. **Supply value** (e.g. number of physicians in an office or tract)
3. **Potential need/demand** (e.g. population in each tract).
4. **Geographic impedance**. Two inputs for each model: (i) travel time/distance between demand and supply and (ii) distance function (e.g. distance decay).
SECTION 3: MODEL DESCRIPTIONS
## FLOATING CATCHMENT AREA MODELS

<table>
<thead>
<tr>
<th>Model</th>
<th>Access Equation</th>
<th>Interpretation</th>
<th>Distance function</th>
<th>Data inputs</th>
<th>Assumptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard FCA</td>
<td>$A_i^{fca} = \frac{S_i}{D_i}$</td>
<td>Average number of providers available to demand (PTD) within an area centered at a population location ($i$) e.g.: physicians to patients ratio, given a commuting distance.</td>
<td>The distance of each buffer around a location ($i$) could take the form of a fixed Euclidean distance (circled area around a tract centroid with a given radius) or travel time defined by the researcher.</td>
<td>Number of providers (e.g. number of physicians) and their locations. Population (or demand) and their location. If the former is used, it is defined as potential access. If information of demand is used, the accessibility score would be interpreted as realized access score.</td>
<td>Population is willing to travel within a given distance or time from location $i$. Supply is inaccessible outside the specified buffer. Every point within the defined buffer is equally accessible. Demand is often approximated by population in a given location.</td>
</tr>
</tbody>
</table>

### Notation

- $A_i^{fca}$: accessibility index; $S_i$: aggregated supply within a distance circle or time-based buffer around location $i$; $D_i$: aggregated demand (or population) at location $i$.  
- *PTD: provider-to-demand ratio*
## FLOATING CATCHMENT AREA MODELS

<table>
<thead>
<tr>
<th>Model</th>
<th>Access Equation</th>
<th>Interpretation</th>
<th>Distance Function</th>
<th>Data Inputs</th>
<th>Assumptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>2SFCA</td>
<td>( A_{2sfca}^{l} = \sum_{i \in T} R_{l} )</td>
<td>Aggregated ratios of PTD* centered at provider locations (l) falling within the catchment area of demand origin (i)</td>
<td>The distance of each buffer around a location i and l (( t_{i} ) and ( t_{l} ), respectively) could take the form of a fixed Euclidean distance (circled area around i and l with a given radius) or travel time defined by the researcher.</td>
<td>Number of providers (or supply) and their locations.</td>
<td>Population is willing to travel within a given distance or time around location i. Provider is inaccessible outside catchment areas. Every point within the defined buffer is equally accessible. However, accessibility to each provider depends on their corresponding demand. Demand is often approximated by population in a given location.</td>
</tr>
</tbody>
</table>

### Notation

- \( A_{2sfca}^{l} \): accessibility index
- \( R_{l} \): ratio within a catchment area of each provider location l
- \( s_{l} \): aggregated supply within a distance circle or time-based buffer centered at provider location l
- \( d_{l} \): aggregated demand (or population) within a buffer centered at location l
- \( T_{l} \): buffer of travel time or fixed distance of each location l.

*PTD: provider-to-demand ratio
# Floating Catchment Area Models

<table>
<thead>
<tr>
<th>Model</th>
<th>Access Equation</th>
<th>Interpretation</th>
<th>Distance Function</th>
<th>Data Inputs</th>
<th>Assumptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>E2SFCA</td>
<td>[ A_i^{e2sfca} = \sum_l R_l W(t_{rl}) ]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step 2:</td>
<td>Weighted sum of PTD ratios centered at provider locations (l) falling within the catchment area of each resident location (i). In contrast to 2SFCA, the aggregated demand of each provider location l is weighted by its distance or travel time within the catchment area (W(t_{rl})). Likewise, the aggregation of PTD are sensitive to distance weights.</td>
<td>The catchment area is defined as in FCA and 2SFCA. However, the weight function, which depends on distance, is defined by a Gaussian function W(t_{rl}). Gaussian functions drop progressively first and then drop sharply as distance increases from an origin. Each catchment area is split into multiple travel time (or distance) zones (often three to four).</td>
<td>Number of providers (or supply) and their locations. Population (or demand) and their location. Distance decay function.</td>
<td>Population is willing to travel within a given distance or time around location i. Providers are inaccessible outside the intersection providers of buffers around a given location. Every point within a defined buffer is not equally accessible. More distant areas are relatively less accessible.</td>
<td></td>
</tr>
</tbody>
</table>

## Notation

- \( A_i^{e2sfca} \): accessibility index; \( R_l \): PTD ratio within a catchment area of each provider location l; \( s_l \): aggregated supply within a distance circle or time-based buffer centered at provider location l; \( d_l \): aggregated demand (or population) within a buffer centered at location l. \( T_l \): buffer of travel time or fixed distance of each location l; \( W(t_{rl}) \): distance weight for each rl travel time defined using a Gaussian function.

*PTD: provider-to-demand ratio
## FLOATING CATCHMENT AREA MODELS

<table>
<thead>
<tr>
<th>Model</th>
<th>Access Equation</th>
<th>Interpretation</th>
<th>Distance Function</th>
<th>Data Inputs</th>
<th>Assumptions</th>
</tr>
</thead>
</table>
| 3SFCA   | Step 3: \( A^3_{fca} \)  
\[ A^3_{fca} = \sum_l G_{rl} R_l W(t_{rl}) \]  
Where  
Step 2:  
\[ R_l = \frac{s_l}{\sum_r d_r G_{rl} W(t_{rl})} \]  
Step 1:  
\[ G_{rl} = \frac{W(t_{rl})}{\sum_l W(t_{rl})} \]  
Weighted sums of PTD ratios centered at providers’ locations (l) falling within the catchment area of each resident location (i).  
In contrast to E2SFCA, the aggregated demand of each physician location l is weighted by the so-called selection weight \( G_{rl} \), or willingness to travel within a buffer, in addition to the impedance weight \( W(t_{rl}) \).  
The catchment area is defined as in FCA, 2SFCA and 3SFCA. However, the weight functions, which depend on distance, are sensitive to both (i) the willingness of each population to travel within a buffer and (ii) impedance (or conflict) from providers to population locations.  
Each catchment area is split into multiple travel time (or distance) zones (often three to four). | Number of providers/supply and their location.  
Population/demand and their location.  
Distance decay function. | Every point within a defined buffer is not equally accessible. Further areas are relatively less accessible.  
The population has a limited area of accessibility.  
Demand to an accessible supplier location is influenced by travel costs (or distance) to that location as well other competing locations. |

### Notation

- \( A^3_{fca} \) : accessibility index
- \( R_l \) : PTD ratio within a catchment area of each provider location l
- \( s_l \) : aggregated supply within a distance circle or time-based buffer centered at provider location l
- \( d_r \) : aggregated demand (or population) within a buffer centered at location l
- \( T_l \) : buffer of travel time or fixed distance of each location l
- \( W(t_{rl}) \) : distance weight for each rl travel time defined using a Gaussian function
- \( G_{rl} \) : selection weight derived from a Gaussian function.
<table>
<thead>
<tr>
<th>Model</th>
<th>Access Equation</th>
<th>Interpretation</th>
<th>Distance Function</th>
<th>Data Inputs</th>
<th>Assumptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAAM</td>
<td>( A_{i}^{raam} = \frac{\sum_{r} d_{rl} / s_{l}}{\rho} + \frac{t_{rl}}{\tau} )</td>
<td>Minimum cost of accessibility at each resident location (i). The cost is measured as the sum of demand to provider ratio (as opposed PTD in previous models) and travel cost time to provider location (l).</td>
<td>The catchment area is defined as in FCA and 2SFCA. However, the weight function, which depends on distance, is defined by a Gaussian function ( W(t_{rl}) ). Gaussian functions drop progressively first and then drop sharply as distance increases from origins. Each catchment area is split into multiple travel time (or distance) zones (often three to four).</td>
<td>Number of physicians (or supply) and their location. Population (or demand) and their location. Distance decay function</td>
<td>Population could potentially access providers at any point within their catchment area. Further distances are avoided if close locations are relatively less expensive in terms of travel costs and congestion costs. The distance dependence responds dynamically to the distribution of supply and demand.</td>
</tr>
</tbody>
</table>

**Notation**

\( A_{i}^{RAAM} \): accessibility index; \( s_{l} \): aggregated supply within a distance circle or time-based buffer centered at provider location \( l \); \( d_{r} \): aggregated demand (or population) within a buffer centered at location \( l \); \( \rho \) and \( \tau \): scale of costs.
### ACCESS SCORE

<table>
<thead>
<tr>
<th>Model</th>
<th>Access Equation</th>
<th>Interpretation</th>
<th>Distance Function</th>
<th>Data Inputs</th>
<th>Assumptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access score</td>
<td>$A_i^{access} = \sum_l W(t_{ri})$</td>
<td>Aggregation of the impedance weight to access to providers within the catchment area of population location (i). e.g.: weighted sum of number of parks within a reasonable catchment area of a population.</td>
<td>The distance of each buffer around a location $i$ ($t_i$) could take the form of a fixed Euclidean distance (circled area around i with a given radius) or travel time defined by the researcher. Within the catchment area, the weights are defined by the distance decay function, as in previous model.</td>
<td>Location of providers (e.g. public parks). Population (or demand) and their location. Distance decay function</td>
<td>Population is willing to travel within a given distance or time around location $i$. Supply is inaccessible outside a defined buffer. Providers do not have restrictions. And willingness to travel to a providers solely depends on its relative distance. Demand is often approximated by population in a given location.</td>
</tr>
</tbody>
</table>

### Notation

$A_i^{ica}$: accessibility index; $W(t_{ri})$: distance weight for each $ri$ travel time defined by a distance decay function.
SECTION 4: VISUAL AND NUMERICAL EXAMPLES
The following model illustrations are based on these characteristics:

- There are two providers: S1 and S2.
- Each provider meets the demand of, at most, 25 people.
- There are 15 tracts.
- The number in the circles represents the population (demand) in each tract (at the tract centroid).
- The maximum time that a person is willing to travel to access a provider is 30 min.
- The buffers around the providers represent the catchment areas of the providers within an assumed travel time. Population centroids within these areas represent the number of people willing to travel to the provider within the catchment area.
STANDARD FCA

1. Visualization
2. Numerical example
Access: Stage 1 – Calculate the Supply/Demand Ratio for Each Location

e.g. the # of physicians per nearby population for each tract
Description:

1. Population located in Tract 1 is willing to travel 30 minutes to access providers (S). In the graph, the catchment area is delimited by the buffer. We assume that population is equal to demand here.
2. S: suppliers locations.
3. The numbers in circles show the population located in each tract.
4. Each S could serve 25 people.

Thus, the accessibility of Tract 1 is given by:

\[
A_{FCA}^{Tract1} = \frac{S1 + S2}{\text{pop(Tract1)} + \text{pop(ii)} + \text{pop(iii)} + \text{pop(iv)}} = \frac{25 + 25}{10 + 40 + 50 + 100} = \frac{50}{200} = \frac{1}{4} = 0.25
\]

where \(\text{pop(i)}, \text{pop(ii)}, \text{and pop(iv)}\) are the corresponding population counts in catchment area of Tract 1.
TSFCA

1. Visualization of stage 1
2. Visualization of stage 2
3. Numerical example
2SCFA: Stage 1 – Calculate the Supply/Demand Ratio for Each Supplier

e.g. the # of physicians per nearby population for each doctor’s office

\[ R_{S1} = \frac{s_1}{\text{pop}} = \frac{25}{310} \]

\[ R_{S2} = \frac{s_2}{\text{pop}} = \frac{25}{300} \]
2SCFA: Stage 2 – Aggregate Nearby Ratios from Stage 1 for Each Origin
e.g. sum the physician-to-pop ratio for all offices within a travel distance to origin Tract 1

\[ A = R_{S1} + R_{S2} = 0.164 \]
Estimation of Accessibility: 2SFCA

**Stage 1:**
Stage 1 of 2SCFA corresponds to the estimation of the catchment accessibility for each provider $S$ and is given by:

$$R_{S1} = \frac{S1}{Pop(S1)} = \frac{25}{200 + 100 + 10} = \frac{25}{310} = 0.08$$

$$R_{S2} = \frac{S2}{Pop(S2)} = \frac{25}{100 + 100 + 40 + 10} = \frac{25}{300} = 0.083$$

where $Pop(S1)$ and $Pop(S2)$ are population counts within the catchment area of $S1$ and $S2$, respectively.

**Stage 2:**
Stage 2 of 2SCFA aggregates the ratios $R$ that lie within the area that the population in a Tract 1 is willing to travel to. In this case, the accessibility of Tract 1 would be given by:

$$A_{1}^{2SFCA} = R_{S1} + R_{S2}$$

$$A_{1}^{2SFCA} = \frac{S1}{Pop(S1)} + \frac{S2}{Pop(S2)} = \frac{25}{310} + \frac{50}{300} = 0.164$$
E2SFCA

1. Visualization of stage 1
2. Visualization of stage 2
3. Numerical example
E2SCFA: Stage 1 – Calculate the Supply/Weighted Demand Ratio for Each Supplier
e.g. the # of physicians per nearby weighted population (based on the zone where they are) for each doctor’s office

\[ R_{S1} = \frac{s1}{\text{Weighted pop}} \]

\[ R_{S2} = \frac{s2}{\text{Weighted pop}} \]
E2SCFA: Stage 2 – Aggregate Nearby Ratios from Stage 1 for Each Origin

E.g. sum the physician-to-pop ratio from stage 1 for all offices within a travel distance to origin Tract 1

\[ A = R_{S1} + R_{S2} = 0.1983 \]
Distance decay function

Explanation based on: https://github.com/GeoDaCenter/spatial_access/blob/master/docs/notebooks/spatial_access_documentation081219.pdf
Stage 1:
Stage 1 of E2SCFA estimates the catchment accessibility for each provider \(S\), as described in 2SCFA. However, the aggregated demand for each provider is weighted by a step function that depends on the distance of the population to \(S\). In this case, the catchment areas are divided into three zones: \(Z1\), \(Z2\) and \(Z3\). Therefore, in the first graph, Tract 1 is in zone 2 (\(Z2\)) of the catchment areas of \(S1\) and \(S2\). Assuming that the corresponding weights of \(Z1\), \(Z2\), and \(Z3\) are 1, 0.75 and 0.5, the accessibility at each \(S\) would be given by:

\[
A1 = \left(0.33 \times 1\right) + \left(0.5 \times 0.75\right) + \left(0 \times 0.5\right) = 0.25
\]

where \(W_{Z1}\), \(W_{Z2}\) and \(W_{Z3}\) are the weights of Zone 1, Zone 2 and Zone 2, respectively. \(Pop1\), \(Pop2\) and \(Pop3\) are the population in each zone of the corresponding catchment area of providers \(S1\) and \(S2\). For instance, \(Pop2\) in \(R_{S1}\) is the sum of 100 and 10 (110), which is the population in Zone 2 of catchment area of \(S1\).

Stage 2:
Stage 2 of E2SCFA aggregates the ratios \(R\) that lie within the area that population in a Tract 1 is willing to travel to. In this case, the accessibility of Tract 1 would be given by:

\[
A_{1}^{E2SFCA} = R_{S1} + R_{S2}
\]

\[
A_{1}^{E2SFCA} = \frac{25}{282.5} + \frac{50}{227.5} = 0.1983
\]
3SFCA

1. Visualization of stage 1
2. Visualization of stage 2
3. Visualization of stage 3
4. Numerical example
3SCFA: Stage 1 – Calculate the selection weight for each tract

e.g. the probability of Tract 1 to select each supplier, given the zone the office is in. If there is only one supplier within the travel distance, then the selection weight equals 1.

\[ G_1 = \frac{w_1}{w_1 + w_2} = 0.66 \]
\[ G_2 = \frac{w_2}{w_1 + w_2} = 0.33 \]
3SCFA: Stage 1 – Calculate the selection weight for each tract
e.g. If there is only one supplier within the travel distance, then the selection weight equals 1, which is the case for all the tracts different from Tract 1, as shown.
3SCFA: Stage 2 – Calculate the Supply/Weighted Demand Ratio for Each Supplier

e.g. the # of physicians per nearby weighted population (selection weight and accessibility weight) for each doctor’s office

\[
R_{S1} = \frac{s_1}{\text{Weighted pop}}
\]

\[
R_{S2} = \frac{s_2}{\text{Weighted pop}}
\]
3SCFA: Stage 3 – Aggregate Nearby Ratios from Stage 2 for Each Origin

e.g. sum the physician-to-pop ratio from stage 2 for all offices within the catchment area of origin Tract 1

\[ A = R_{S1} + R_{S2} = 0.2016 \]
Stage 1:

The first stage of 3SCFA determines the selection weight of each provider location for each tract. In this example, S2 is within Zone 3 of Tract 1 and S1 is in Zone 1 of Tract 1. Assuming the same Zone weights used in E2SCF, the selection weights of the population in Tract 1 are:

\[
g_{S1}^{Tract1} = \frac{g_{S1}^1}{g_{S1}^1 + g_{S2}^1} = \frac{1}{1 + 0.5} = 0.66
\]

\[
g_{S2}^{Tract1} = \frac{g_{S2}^1}{g_{S1}^1 + g_{S2}^1} = \frac{0.5}{1 + 0.5} = 0.33
\]

Where \(g_{S1}^1\) and \(g_{S2}^1\) are the weights of selecting S1 and S2, respectively, for the population located in Tract 1.

For simplicity, we are assuming that other locations can only choose one supplier, as shown in the second graph. Except for Tract 1, most buffers only include one provider. In other words, their selection weights is always 1. They cannot choose more than one provider, given that their catchment areas have, at most, one provider. It is: \(g_{S1}^{Tract1}\) and \(g_{S2}^{Tract1}\) are 1 and 1, respectively.
Stage 2:

Stage 2 of E2SCFA estimates the catchment accessibility for each provider $S$, as described before. However, the aggregated demand for each provider is weighted by a step function that depends on the distance of the demand to $S$ and the selection weights estimated in the previous stage (G1 and G2). In this case, the accessibility to each location $S$ is:

$$R_{S1} = \frac{S1}{(Pop(vi) \times W_{z1} \times G_{S1}^{vi}) + (Pop(vii) \times W_{z2} \times G_{S1}^{vii}) + (Pop(Tract1) \times W_{z2} \times G_{S1}^{Tract1})}$$

$$= \frac{25}{(200 \times 1 \times 1) + (100 \times 0.75 \times 1) + (10 \times 0.75 \times 0.66)} = 0.0893$$

$$R_{S2} = \frac{S2}{(Pop(ii) \times W_{z1} \times G_{S1}^{ii}) + (Pop(iii) \times W_{z2} \times G_{S2}^{vii}) + (Pop(iv) \times W_{z2} \times G_{S2}^{iv}) + (Pop(v) \times W_{z3} \times G_{S2}^{v}) + (Pop(Tract1) \times W_{z2} \times G_{S2}^{Tract1})}$$

$$= \frac{25}{(50 \times 1 \times 1) + (100 \times 0.75 \times 1) + (100 \times 0.75 \times 1) + (40 \times 0.5 \times 1) + (10 \times 0.75 \times 0.33)} = 0.112372$$

Where:

$G_{S1}^{vi}$: selection weight of Tract vi of choosing S1, $G_{S1}^{vii}$: selection weight of Tract vii of choosing S1.

$W_{z1}, W_{z2}, W_{z3}$: weights of Zone 1, Zone 2 and Zone 3, respectively.
Stage 3:

Stage 3 of 3SCFA consists of the aggregation of the individual suppliers’ rates estimated in the previous stage that lie within the catchment area of a demand location. In this case, the accessibility of Tract 1 would be given by:

\[ A^{3SCFA}_1 = R_{S1} + R_{S2} \]

\[ A^{2SFCA}_1 = 0.0893 + 0.112372 = 0.2016 \]
ACCESS: WEIGHTED CATCHMENT
Access: Stage 1 – Calculate accessibility ratio based on distance to each supplier
e.g. access cost of Tract 1 of selecting each doctor’s office given the zone the office is in
Description:

1. Population located in **Tract 1** is willing to travel 30 minutes to access providers (S). In the graph, the catchment area is delimited by the buffer. We assume that population is equal to demand here.
2. **S**: supplier location.
3. The numbers within circles show the population located in each tract.
4. Each S could serve 25 people.

Thus, the accessibility of **Tract 1** is given by:

\[
A_1^{access} = 1 + 0.5
\]

\[
A_1^{access} = 1.5
\]
1. Visualization of stage 1
2. Visualization of stage 2
3. Visualization of stage 3
4. Numerical example
RAAM: Stage 1 – Each tract selects a provider, given others’ selection

e.g. below, Tract 1 could choose S1 or S2, while other tracts only can access one provider
RAAM: Stage 1 – Each tract selects a provider

e.g. below, Tract 1 could choose S1 or S2, while other tracts only can access one provider
RAAM: Stage 2 – Evaluates outcome from hypothetical scenarios

e.g. Tract 1 evaluates the costs of choosing S1, given others’ decisions

\[ A = \frac{310}{25} + \frac{5}{30} = 12.5 \]

Demand of S1: 310
Demand of S2: 290

Tract centroids with pop. count (demand)

Centroids with pop. count – no nearby supplier

Tract 1: Origin tract

Suppliers’ locations for which ratios are calculated

Area with 30 min access from centroid to S1

Supplier chosen by each tract

Tract 1 decision
RAAM: Stage 2 – Evaluates outcome from hypothetical scenarios

e.g. Tract 1 evaluates the costs of choosing S2, given others’ decisions

Demand of S1: 300

Demand of S2: 300

$A = \frac{300}{25} + \frac{20}{30} = 12.7$

Supplier chosen by each tract

Tract 1 decision

Tracts

Centroids with pop. count – no nearby supplier

Tract centroids with pop. count (demand)

Tract 1: Origin tract

Suppliers’ locations for which ratios are calculated

Area with 30 min access from centroid to Tract 1

Supplier chosen by each tract
The Rational Agent Access Model (RAAM) generates a spatial access score for each point of origin, like a tract, that indicates how spatially accessible that tract is to services. Like with the FCAs, the travel time to a provider is take into account here. But in addition, a measure of service saturation (congestion) is also included in RAAM. Access is defined as the sum of congestion and travel. Congestion is the ratio of demand over supply, normalized by the area mean, e.g. patients over doctors, scaled by the area average. Travel time is the time it takes to travel from e.g. a tract to a doctor. This is normalized by the time patients are willing to travel, e.g. a 45 min drive. Patients seek to minimize the cost of congestion at a provider and the travel time to the provider. This is solved with a greedy optimization algorithm that iterates over the points of origin.
Estimation of Accessibility: RAAM

The arrows in the graphs reflect the decision to choose a certain supplier. Each tract’s population iteratively evaluates its decision which provider to choose by minimizing travel time to and service congestion at each location. **Tract 1**’s decision to select either **S1** or **S2** is expressed as the minimization of its costs of going to **S1**, given that **S1** meets the demand of 300 people from 2 more tracts that would select **S1** -- versus **S2** that meets the demand of 290 people from 4 other tracts.

The decision problem of **Tract 1** is then given by:

\[
\text{Min} \left\{ A_1^{RAAM}, A_2^{RAAM} \right\} = \text{Min} \left\{ \frac{\text{Pop}_{S1}}{S1} + \frac{t_{s1}}{t}, \frac{\text{Pop}_{S2}}{S2} + \frac{t_{s2}}{t} \right\} = \\
\text{Min} \left\{ \frac{200 + 100 + 10}{25} + \frac{5}{30}, \frac{100 + 100 + 50 + 40 + 10}{25} + \frac{20}{30} \right\} \\
\text{Min}\{12.5, 12.7\} = 12.5
\]

where \(\text{Pop}_{S1}\) and \(\text{Pop}_{S2}\) are the demand of **S1** and **S2**, given the hypothetical decision of **Tract 1** to go to **S1** and **S2**, respectively. \(t_{s1}\) and \(t_{s2}\) are the travel times from **Tract 1** to **S1** and **S2**, respectively. \(A_1^{RAAM}\) and \(A_2^{RAAM}\) are the costs of choosing **S1** and **S2**, respectively. 5 and 20 are the minutes that it takes from **Tract 1** to visit **S1** and **S2**, respectively. 30 is a common number chosen for standardization purposes.

Given that costs are minimized for **Tract 1** if it chooses **S1**, the accessibility is then given by:

\[
A_1^{RAAM} = \frac{200 + 100 + 10}{25} + \frac{5}{30} = 12.5
\]
## COMPARISON OF MODELS:
Is Calculation Centered on Origin or Destination (for each Stage)?

<table>
<thead>
<tr>
<th></th>
<th>FCA</th>
<th>2SFCA</th>
<th>E2SFCA</th>
<th>3SFCA</th>
<th>Access</th>
<th>RAAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>STAGE 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Origin</td>
<td><img src="image1" alt="Image" /></td>
<td><img src="image2" alt="Image" /></td>
<td><img src="image3" alt="Image" /></td>
<td><img src="image4" alt="Image" /></td>
<td><img src="image5" alt="Image" /></td>
<td><img src="image6" alt="Image" /></td>
</tr>
<tr>
<td></td>
<td><img src="image7" alt="Image" /></td>
<td><img src="image8" alt="Image" /></td>
<td><img src="image9" alt="Image" /></td>
<td><img src="image10" alt="Image" /></td>
<td><img src="image11" alt="Image" /></td>
<td><img src="image12" alt="Image" /></td>
</tr>
<tr>
<td>STAGE 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Origin</td>
<td><img src="image13" alt="Image" /></td>
<td><img src="image14" alt="Image" /></td>
<td><img src="image15" alt="Image" /></td>
<td><img src="image16" alt="Image" /></td>
<td><img src="image17" alt="Image" /></td>
<td><img src="image18" alt="Image" /></td>
</tr>
<tr>
<td></td>
<td><img src="image19" alt="Image" /></td>
<td><img src="image20" alt="Image" /></td>
<td><img src="image21" alt="Image" /></td>
<td><img src="image22" alt="Image" /></td>
<td><img src="image23" alt="Image" /></td>
<td><img src="image24" alt="Image" /></td>
</tr>
<tr>
<td>STAGE 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Origin</td>
<td><img src="image25" alt="Image" /></td>
<td><img src="image26" alt="Image" /></td>
<td><img src="image27" alt="Image" /></td>
<td><img src="image28" alt="Image" /></td>
<td><img src="image29" alt="Image" /></td>
<td><img src="image30" alt="Image" /></td>
</tr>
<tr>
<td></td>
<td><img src="image31" alt="Image" /></td>
<td><img src="image32" alt="Image" /></td>
<td><img src="image33" alt="Image" /></td>
<td><img src="image34" alt="Image" /></td>
<td><img src="image35" alt="Image" /></td>
<td><img src="image36" alt="Image" /></td>
</tr>
</tbody>
</table>
THANK YOU

Jamie Saxon
Dan Snow
Logan Noel
Irene Farah
Luc Anselin
Xun Li
Karina Acosta

Julia Koschinsky
Larissa Vieira
George Oliver
Yair Atlas
Bryan Wang
Caitlyn Tien
Richard Lu