Competing destination for Spatial Interaction models, SpInt/PySal

Project Proposal for Google Summer of Code 2021

Personal Information

- **Full Name:** Lenka Hasova
- **Current Education:** 2nd year PhD in Advanced Quantitative Methods at University of Bristol, UK
- **Email:** haska.lenka@gmail.com, lenka.hasova@bristol.ac.uk
- **Phone:** +4407305328908
- **GitHub:** https://github.com/lenkahas
- **Timezone:** Central European Time GMT + 1

Abstract

Competing Destination (CD) models, which are an extension of Spatial Interaction (SI) [1] models, has been around since 1980's [2] and are often used within Economic and Social Sciences [3] [4]. CD models involve the analysis of flows from an origin to a destination similarly to traditional SI models, however, CD includes the 'Competing Destination' term also known as 'Accessibility' term, which accounts for the spatial-structure effect of the flows from behavioural perspective.

Although the SI models are established within Python and R modules already (SpInt, simR), specification for deriving Accessibility term for CD estimation is missing. This project aims to fill this gap by developing a Competing Destination class that will include the computation of the accessibility term and will be binded to the existing SpInt module. The main challenge here is the scaling of the accessibility computation for large datasets. This is an important aspect of the project as flow datasets often have hundred thousands and even millions records.

This project will extend the use of the SpInt module and PySAL library for high level analysis of spatial flows and will be a valuable addition to spatial analysis in python.

---

[1] Wilson, A. G. (1971) 'A Family of Spatial Interaction Models, and Associated Developments', Environment and Planning A: Economy and Space, 3(1), pp. 1–32. doi: 10.1068/a030001.

[2] Fotheringham AS. A New Set of Spatial-Interaction Models: The Theory of Competing Destinations. Environment and Planning A: Economy and Space. 1983;15(1):15-36. doi:10.1177/0308518X8301500103

[3] Felipa de Mello-Sampayo, 2017. "Competing-destinations gravity model applied to trade in intermediate goods," Applied Economics Letters, Taylor & Francis Journals, vol. 24(19), pages 1378-1384, November.

[4] Mattia Cattaneo & Paolo Malighetti & Michele Meoli & Stefano Paleari, 2017. "University spatial competition for students: the Italian case," Regional Studies, Taylor & Francis Journals, vol. 51(5), pages 750-764, May.

Project plan

The project has 5 phases.

Phase/Date	Name	Task	Done	Deadline
-1: Before project start	Establishing the background	Exploring the CD theory and establishing the steps for computation of accessibility term	☑	June 1
		Build basic function calculating accessibility term	☑	June 1
		Test its functionality on differently sized dummy data	☑	June 1
0: Community Bonding Period	Establishing the connection and material	Establish repository	Partially	June 1
		Get to know the mentors	☐	June 7
		Get familiar with package building, reviewing and testing	☐	June 7
		Get familiar with SpInt	☐	June 7
1: June 7 - July 11	Developing the main Code	Review established code	☐	June 14
		Discuss the methods of scaling up and establish the steps for building the main code	☐	June 14
		Writing main code Start	☐	-
		Progress discussion	☐	July 26
		Continue writing code	☐	-
		Start developing the tests	☐	-
		Blog_1	☐	July 11
2: July 16 - August 16	Testing and debugging	Running tests - Run_1	☐	July 25
		Debugging and discussing next steps	☐	August 01
		Running tests - Run_2	☐	August 16
		Blog_2	☐	August 16
3: August 16 - August 23	Submitting the work	Submitting the final product	☐	August 22
		Create report + Blog_3	☐	August 31

Further Details:

Deliverables:

- Competing Destination class for SpInt - Phase 1
- Tests for the class, reviewed class - Phase 2
- Blog Posts & Report - Across the phases

Tools:

- Github Repository - https://github.com/lenkahas
- Blog - https://www.lenkahas.com

What do we already have:

Phase -1, establishing the knowledge of CD theory, writing the base function for accessibility term and testing on dummy data, is more or less completed. This can be found in this repo, together with some early comments from one of the mentors (Levi J. Wolf).
The accessibility term is (from the initial code) calculated by;
For each row of the flow data

1. Define the origin O and destination D node
2. Find all possible destinations to origin O
3. Construct flows from destination D to all possible destination to origin O
4. From those remove the the flow O to D
5. Calculate the SUM(distance* destination mass)
6. Append the result to array
7. Transfer array to column in flow data

What do we need:

- Figure out how to rewrite this function to be faster (vectorization, eliminating unnecessary processes, reduce memory footprint, move calculations out of loop,...)
- Incorporate this into the count model from SpInt
- Define class calculating the CD term
- Define test to check its functionality
- Connect class to gravity model in SpInt

Development Experience

You can find me on GitHub as 'lenkahas'. I have been previously involved in Bristol Solar Panel Potential project which calculated the solar potential of Bristol roofs using LiDAR data, gdal command tools and R. The code and the results for this project are save in GitHub repository. I also deploy my blog using Github and Netlify. I use the blog to post reproducible tutorials for methods I use in my research.

Why this project?

This project is directly related to my PhD research that investigates the Application of Advanced Quantitative Methods to Spatial interaction models. As a young scientist in 21th century I learn not just from journals and books but also from a code developed to test the theories. In my research, I tried to recreate the Competing Destination model, which

happened to be much harder than expected, mainly because there is no software to accommodate these models in my research. By building and incorporating the CD model into existing python library will, firstly, create a better understanding of the underlying processing for the CD specification of SIM, and secondly, it will allow other researchers to use the framework for their own research and build future scientific knowledge. Additionally, I believe an experience in open source software development will be beneficial, maybe even essential, for my current and future research.