Corrine Rojas

GIS 441

Requirements Analysis Document

Professor Serge Rey

Spatial Dynamics: Space-Time Interaction Tests

1. Introduction

1. Purpose of your package

The purpose of the package is to add Space-Time Interaction Tests to the PYSAL plugin functionality. This will allow users to be able to use more tools within the set of spatial dynamics, which exist under the umbrella of the PySAL toolkit.

2. Scope of your package

The scope of the package will be adding the default Knox Test and Modified variations to the new Space-Time Interaction Tests toolkit of the PySAL plugin.

3. Objectives and success metrics for your project

The objective and metric used to ensure success are to make a tool that will accurately run an interaction test with accurate visualization according to the statistical models supplied by PySAL.

4. Definitions, terms

a. Space-time Interaction Tests: A series of tests to detect spatial clustering with space-time event patterns using any “purely spatial or temporal clustering.” This means that only data sets with a time variable will be able to be used in these analyses.

b. Knox Test: One of the most common interaction tests, it employs a user-defined “critical threshold” in space and time variables to define “proximity” between events (Rey 2013).

c. Modified Knox Test: A variant on the Knox Test, here we find a feature that measures the difference between the original observed Knox statistic and its expected value (Rey 2013).

d. Mantel Test: In this test, we take a sum of the product of the spatial and temporal distances between all event pairs, which is how it differs from the Knox test (instead of discarding information regarding event distances) (Rey 2013).

e. Jacquez Test: In this test, we employ a nearest neighbor approach to calculating distance instead of a set distance in time and space proximity (Rey 2013).

5. References

Rey, S. (2013). Space-time interaction tools. Getting Started with PySAL: Spatial

Dynamics. Retrieved from <http://pysal.readthedocs.io/en/latest/users/tutorials/dynamics.html#id20>

6. Overview

As stated above, the purpose of the package is to add Space-Time Interaction Tests to the PySAL plugin functionality to develop the Spatial Dynamics tool even further. Users will be able to use more statistical models and their respective visualizations within the set of spatial dynamics, which exist under the umbrella of the PySAL toolkit.

2. Current System

Currently, there is functionality of this PySAL Processing in QGIS plugin that exists. The toolkit relies on shapefiles provided by the user. To extend existing work, I will need to use the same customs and statistical methods already provided by the PySAL project. I will introduce the necessary “space-time” variable to run the Space-Time Interaction analyses. I suspect there is more integrative hurdles to overtake as I further develop the project, but this is one definite change that the existing system can easily support.

3. System Proposal

1. Overview

This proposal is to implement the Spatial Dynamics Space-Time Interaction Tests to the PySAL plugin. This will increase user functionality of the PySAL Processing toolkit. Users will be able to use more statistical models and their respective visualizations within the set of spatial dynamics, which exist under the umbrella of the PySAL toolkit.

2. Functional Requirements

1. Listing of features to be implemented

Space-time Interaction Tests, including the Knox Test, Modified Knox Test, Mantel Test, and Jacquez Test. These tests will be ran in a GUI that matches existing design customs.

2. Mock-ups (sketches) of features

Available at: http://imgur.com/a/XGXeB

3. Nonfunctional Requirements

1. Useability

The GUI will provide a very easy user experience by employing a straightforward workflow, from inputting the shapefile all through picking a proximity threshold (if applicable to the selected interaction test). Further, a short description of each test can be made available in a “help ?” GUI feature.

2. Reliability

Once the main bulk of the code is employed, the test should be reliable as long as the statistical models stay in place.

3. Performance

Besides pre-existing byte memory management issues, the test should be fairly reliable and perform quickly without using too much memory.

4. Supportability

The toolkit will include easily read commented code. QGIS uses Python 2.7, therefore the tests will also be written and processed in Python 2.7. Once QGIS goes towards implementing Python 3, it should be easy to support as well.

5. Implementation

Changes will be implemented by Corrine Rojas.

6. Interface

As mentioned, the processing analyses will be able to be ran through a GUI in QGIS.

7. Packaging

It will be released in the newest version of PySAL Processing Toolkit for QGIS.

8. Licensing

PySAL is an open source project Copyright (c) 2007-2014, PySAL Developers. All rights reserved. QGIS is a user friendly Open Source Geographic Information System (GIS) licensed under the GNU General Public License. QGIS is an official project of the Open Source Geospatial Foundation (OSGeo). It runs on Linux, Unix, Mac OSX, Windows and Android and supports numerous vector, raster, and database formats and functionalities.

4. Project Management

1. Schedule

1. Detailed milestones for project

November 18, 2016 – Presentable GUI that encompasses all test.

November 25th, 2016 – Working statistical models and visualization with inputted shapefile.

December 2nd, 2016 – Quality Assurance and testing

End of first week of December: Presentation and release.

2. Repositories

1. URL for project repository

https://github.com/corrinerojas/processing\_pysal