ReadMe for geo_set_cover iPython notebook

Author: Jeff Nivitanont, GeoCARB Research Assistant

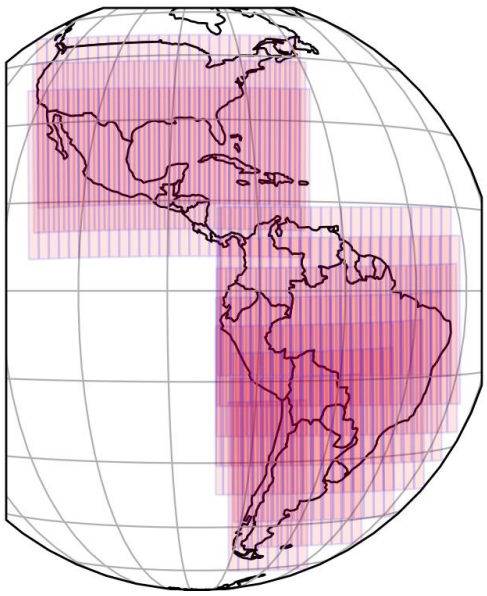
Location: University of Oklahoma

Packages:

- netCDF4 - used to read in netCDF4 formatted files
- NumPy - part of the SciPy suite, provides data structures such as n-d arrays and series
- pandas - Python Data Analysis library provides easy to read and manipulate DataFrame
- shapely - Python wrapper for C-GEOS (Geometry Engine - Open Source), which is a port of the JTS (Java Topology Suite)
- GeoPandas - Geospatial Data Analysis library built from pandas
- matplotlib - Primary plotting library for python
- CartoPy - add-ons for plotting geospatial information in matplotlib

We are given the problem of optimizing the scan pattern for the GeoCARB satellite instrument set to launch in 2022. GeoCARB is a geostationary satellite that will be positioned at 0-85W and make daily observations of atmospheric CO2, CO, NH4, and SIF over the Americas.

The underlying mathematical problem related to optimizing the scanning pattern is called the Geometric Set Cover problem. Given a set of points in 2-D space and a set of covers, what is the optimal covering set? This problem is known to be NP-Hard so there are no efficient methods for straight-forward computation. However, there are heuristic methods for finding a set cover, though they do not necessarily yield optimal covers. We start our exploration into this problem with the Greedy Heuristic.



We use the shapely package to create the scan block Polygons from the coordinates of the corners from the scan blocks indicated in the netCDF4 file. We plot them on our map to the right using the matplotlib and CartoPy libraries.

Using built-in GeoPandas dataset we select North America and South America geometries for our set to be covered. The Data is 1:110m from NaturalEarth and includes cultural and physical information (i.e. countries, pop, geometries, etc.). NaturalEarth data is a free for use and supported by NACIS (North American Cartographic Information Society). After obtaining our geometries, we truncate our set to only include land between 50N and 50S and exclude eastern Nova Scotia, which lies outside our scanning blocks.

We buffer each scan block as a workaround to underlying floating point precision issues in the C GEOS package, which the python shapely package is wrapped around. Although, this won't be an issue with the integration into the instrument since each scan block will overlap the next. The ticket to fix the bug can be found here https://github.com/Toblerity/Shapely/issues/357.

After establishing our universal set to be covered and having our set of subcovers (scan blocks), we are ready to implement the Greedy Heuristic Algorithm for finding covering sets, which is shown on the right. Finally, we obtain a covering set from the algorithm stored in the GeoDataFrame named cover_set plotted in red in the picture below. More details about the code can be found at https://github.com/jeffniv/crowell-lab/blob/master/geo_set_cover-git.ipynb

Algorithm GSC(X,F)
X : A finite set .
F : A collection of subsets of X.
begin
$C \leftarrow \phi$
$W \leftarrow X$
while(W is not empty)
Select $S \in F \setminus C$ that maximizes $|S \cap W$
|
$W \leftarrow W \setminus S$
$C \leftarrow C \cup \{S\}$
end while
return C
end