GEOG205: Lecture 4: From Attribute to Geography — Distance, Neighbourhood, Slope and Aspect

Roger Bivand

January 2002

1 Distance measurement

The last lecture presented an outline of computational processes and how they are incorporated into the scripting languages of GIS. This lecture is concerned with extending our as yet limited toolbox of compound procedures from just manipulating attributes in-place, and overlaying map layers, to make use of our knowledge of where geographical objects are located. We will in turn see how distance and the measurement of distance can be introduced, next how objects close to one another may be related, and finally how to start dealing with surfaces. On Thursday, we will complete our treatment of analysis by bringing in iterative methods for viewsheds, cost accumulation, drainage, and network operations, together with interpolation methods.

Since map layers contain geometric information on where the objects and their attributes are placed, we can use this to answer more complex questions than "what", "where", and combinations of "what" questions, such as "where are cells classified as `inhabited` in the **`population`** layer that are classified as `water` in the **`land cover`** layer". We introduce questions like "how far", "how near", and the "buffer" concept.

1. The spatial reference system and distance measurement
   - In order to measure distance or direction, even on a single map layer, we have to know its spatial reference system.
   - We also have to be aware that assumptions made in projection geometry and choice of geoid will introduce errors into our measurements. Further, if the data we are using are secondary, for instance scanned or digitised from paper maps, the printed representation will also deviate from the original drawings, and may also have been subject to geometric generalisation in draughting.
   - However, since all measurements are approximations, we handle this by giving upper and lower bounds for our results.
   - Finally, in phenomena we are interested in, measurement results will be scale-dependent, because of the fractal nature of many natural phenomena in vegetation and geomorphology, which feed through into landscape studies.
2. Measuring distance in the raster and vector approaches
   - Raster map layers are defined in terms of a regular grid. Given that the grid cells have been assigned dimensions measured say in metres, distances in the grid can be established by counting the number of intervening cells if the distance to be measured follows grid lines.
   - If the required distance cuts across the grid pattern, the measurement line will either be built up of straight lines and diagonals close to the "true" line, or be taken between the centroids of the two cells.
   - Vector systems are designed to measure distances directly in the chosen coordinate system, and are limited by the tolerances of the input hardware rather than any quality of the map layer. Finding attribute values along the transect described by the line drawn is more challenging.
   - Raster systems can do this at little extra cost. Most raster systems "play" at using vector precision for distance measurement, but for analysis they use "queen" movements for each step.
   - Remember that distances are measured in the plane.

3. Buffers and their application

   - Now that we can measure distances between objects, it follows that we can also measure distances around them too. This is known as the buffer operation, and will either yield categories within chosen distances, or a "spread" of increasing distances from the chosen objects to be reclassified or "sliced" later.
   - A typical use of buffers is to determine zones around point or linear objects to which a specific management regime is to be applied. If the buffer is one-sided, it can be termed a "set-back".
   - Buffers can also be constructed around area objects; this essentially involves three steps. The procedure first finds the edge of the object, then buffers out from that as if it were a line, and finally removes the buffer areas in the object's interior.
   - Buffer map layers can be used like all other layers, for example by overlay to establish whether land use categories approach each other too closely. Remember that the distances measured may not be exact, and try to set the buffer distances in the light of the precision of your input map layer.

4. Triangulation and Voronoi diagrams

   - If we have several point objects, and spread distance measurements out from each, we will find that half-way between each pair (by definition) the distances will be equal. This operation of "fencing" is the raster equivalent of the geometric procedure of triangulation.
   - Triangulation starts from a distribution of point objects in the plane, and constructs lines between them. Obviously some lines cross each other, and in these cases the shorter line is retained. This "Delaunay" triangulation defines for each point its set of proximal neighbours. The "Voronoi" or "Dirichlet" tesselation is then found by drawing new lines at 90 degrees to the midpoints of this triangulation, to form polygons. These polygons define the area which is "closest" to the point in the chosen metric.
   - Triangulation can also be used as a procedure for contouring when the points have an attribute value, but it starts off just as a way of partitioning space and finding the set of proximal neighbours for each point.
   - For regularly spaced points in a square grid, the "Voronoi" tesselation is the pattern of grid cells. Each has four proximal neighbours in the "rook" (tårn) pattern.

2 Neighbourhood operations and filtering

Both buffering and triangulation have introduced the notion of proximity, underlining the potential importance of relationships between objects in space, between neighbours. Tobler's first law of geography is that: "Everything is related to everything else, but near things are more related than others". We will see that we can programme neighbourhood procedures by designing filters.

1. Neighbourhoods vector-style

   - Vector GIS with topological storage of geometric data already "know" about their neighbours. Typically, each arc in an arc-node structure will record the ID-number of the polygons on its left and right.
   - Consequently, asking questions like "Find the sum of the lengths of the boundaries between residential and pasture categories in the land cover map layer" is uncomplicated.
   - The same applies to making lists of neighbours: "Make a list of all the owners of properties neighbouring schools in the north west of the city", where neighbouring is taken to mean sharing a boundary in a cadastral map layer. The correctness of the list does however depend on the list of owners being up-to-date, being linked to the correct property ID's, and on the precision of the cadastral map layer.
   - Distance relations are more of a problem, because the system will typically find every object within the specified distance before going back to the database to check whether attribute conditions are fulfilled.

2. Buffers and neighbourhoods

   - In the raster case, the objects “know” nothing about their neighbours — the “knowledge” is simply part of the grid system, and has to be reconstructed each time it is needed. On the other hand, it is regular and uncomplicated.
   - The procedures we have already learnt permit us to answer questions like “Find the sum of the lengths of the boundaries between residential and pasture categories in the land cover map layer”, even though we have no topological measure of the category polygons.
   - One approach is to buffer one of the two categories out by one grid cell. If we then overlay the resulting map over the other category and cross-tabulate, we will find a category with a cell count approximating the boundary between the categories.
   - Naturally, the result will not be precise, but it is unlikely in the first place that metre-precision is a realistic expectation — this is a typically “fuzzy” topic.

3. Filtering and neighbourhoods

   - Filtering is a way of approaching data that comes from modifying unidirectional series, most often in time. In many cases it makes sense to compare results now with results from a comparable month or season previously, rather than with those immediately before. Spatial filters are different because they move simultaneously in two directions across the plane; they are 2D filters.
   - Filtering is much used in processing remotely sensed data, in order to reduce technical imperfections, and to enhance image quality. In vector GIS, filtering is difficult to handle, but in raster systems, it is simply another way of doing overlays.
   - If we start with a 3x3 filter, there are 9 positions: the cell itself, and its 8 neigbours to the N, NE, E, SE, S, SW, W, and NW. Overlay as we know it applies to the cell itself. Using filtering just diplaces the layer in the cardinal direction of the neighbour being processed. Remember that there is no data beyond the edges of the layer.
   - Say we expect the value of each cell to be the average of its neighbours. We can design a filter to calculate this as a little table, using map algebra multiplication. Here is the GRASS filter for “rook’s” average:

     ```
     TITLE 3x3 rook’s move
     MATRIX 3
     0 1 0
     1 0 1
     0 1 0
     DIVISOR 0
     TYPE P
     ```

     This can be used to compute expected values for cells, in this case a measure of spatial autocorrelation — a measure of smoothness in this case.

4. Programming a slope filter

   - Before we move further to discuss surfaces, it may be helpful to see how powerful filters can be. The do not have to just be 3x3, and can be much larger, if the processes hypothesised have a larger range. For our present purposes we will keep to 3x3. We only have a simple raster GIS, which does not have a “slope” or “gradient” operation, but does filter, because filter is just overlay with displacement.
   - A standard method of estimating slope is to fit a 2D plane though the 9 cells in a 3x3 neighbourhood; Chrisman gives the details on p. 166. This describes the same algorithm as Map Factory presents (module reference p. 118).
   - The filters we need are for the two dimensions, X:

     ```
     TITLE 3x3 Sobel x filter
     MATRIX 3
     -1 0 1
     -2 0 2
     -1 0 1
     DIVISOR 8
     TYPE P
     ```

and Y:

```
TITLE 3x3 Sobel y filter
MATRIX 3
-1 -2 -1
0 0 0
1 2 1
DIVISOR 8
TYPE P
```

- Running the following pseudo-code (like a script) gives us a layer with slopes in degrees (after cleaning up edge effects):

```
slope.in.x = Filter elevation using "Sobel x filter";
slope.in.y = Filter elevation using "Sobel y filter";
slope = sqrt((slope.in.x ^ 2) + (slope.in.y ^ 2));
```

3 Surfaces, slopes and aspect

We have just started looking at surfaces, slopes and aspect, but ought to step back a little to impose more order. Many GIS just see elevation as the only important surface — because they are dimensionally challenged. It is also reasonable to apply reference system criteria to elevation surfaces, measuring them in the same metric. Many applications however also deal with continuous surfaces of other attributes than elevation, but here slope and aspect will not share a metric with the spatial reference system (fall in pH per metre of a profile?).

1. The spatial reference system and surfaces

    - Had GIS incorporated three dimensions, slope and aspect would have been part of the topological system on which attributes were registered. As it is, measurements of attributes, including elevation, are made in the 2D plane, followed by assumptions about the kind of surface they belong to.
    - Often systems simplify surfaces, sometimes treating them as piecewise continuous (a raster DEM), sometimes as continuous with abrupt changes in slopes (TIN — triangulated irregular network terrain representation), and sometimes for analysis as continuous with continuous rates of change.
    - We will see next week how interpolation lets us estimate the fit of point-based attribute data to fill a study region, and how this may introduce error into the surfaces we use, both for elevation and other attributes of interest.

2. Slope and aspect procedures for DEM

    - In fact most raster GIS provide compound procedures for computing slope and also aspect, the direction in which the slope is facing. The methods used are often refinements of the filter presented above, avoiding for instance problems in determining aspect when slope is close to zero.
    - It is important to remember that filter-based procedures are subject to edge effects, and that they cannot be expected to estimate slope or aspect better than the underlying algorithm. If, as in our example, slope and aspect are taken from fitting a simple plane to 8 points in 2D, the results will not capture significant non-linearity, and may not tackle summits or depressions well.
    - This is partly because a DEM is a 2D piecewise continuous function, and the filters try to force it to be continuous with abrupt changes in slope. The next step logically is to estimate the slope and aspect from draping a continuous function over all the data, and taking derivatives.