Grouping village points based on distance and population size using ArcMap

Asked 8 years, 1 month ago Modified 1 year, 3 months ago Viewed 2k times

7

How do I group a number of villages based on their distance to each other and total cumulative population size?

I have a vector file of say 100 village points attached to atable containing data on population size and so forth. I want to group these villages such that each group contains 5-6 villages with the cumulative population of size of 5000. Hence, distance and cumulative population size should be used in my clustering of villages. I have used grouping in ArcMap 10.2 with no success. I have ArcMap 10.2.

arcgis-desktop arcmap arcgis-10.2

Share Improve this question Follow

edited Sep 22, 2021 at 0:40
PolyGeo ♦
64.4k 28 102 316

asked Nov 24, 2014 at 20:46
Rassoul Mahiny
81 2

1 It would help us if you had an idea of what villages you are hoping to look at. Otherwise, it is not possible for us to help you, if we can. – Kotebiya Nov 24, 2014 at 21:08

1 Would you be able to edit your question to include the version of ArcGIS for Desktop that you are using and what you have tried so far, please? – PolyGeo ♦ Nov 24, 2014 at 21:16

What sort of number? are there 100 villages total, 1000, 10000000? Semi-manual methods may help for aggregating less than 1k points as operator time would be less than scripting effort, more than that would need more automated methods... scripting time being less than operator time. – Michael Stimson Nov 24, 2014 at 21:32

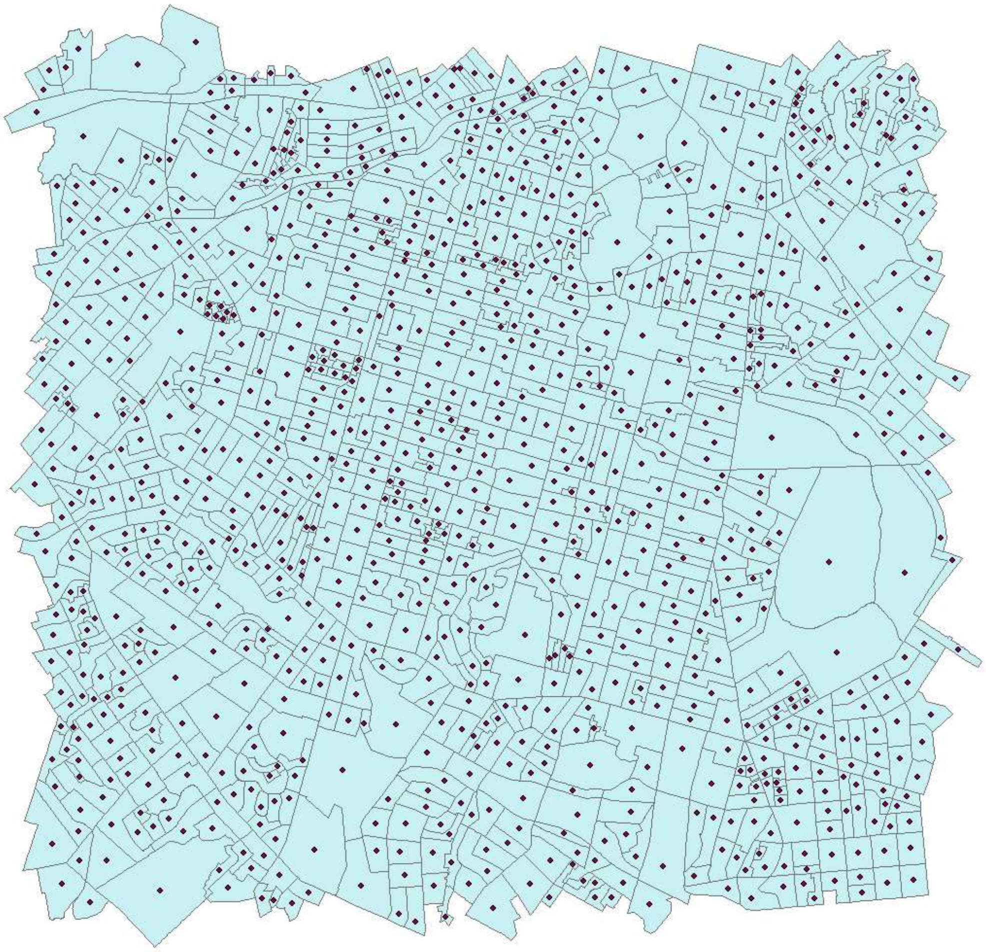
1 Answer

Sorted by: Highest score (default)

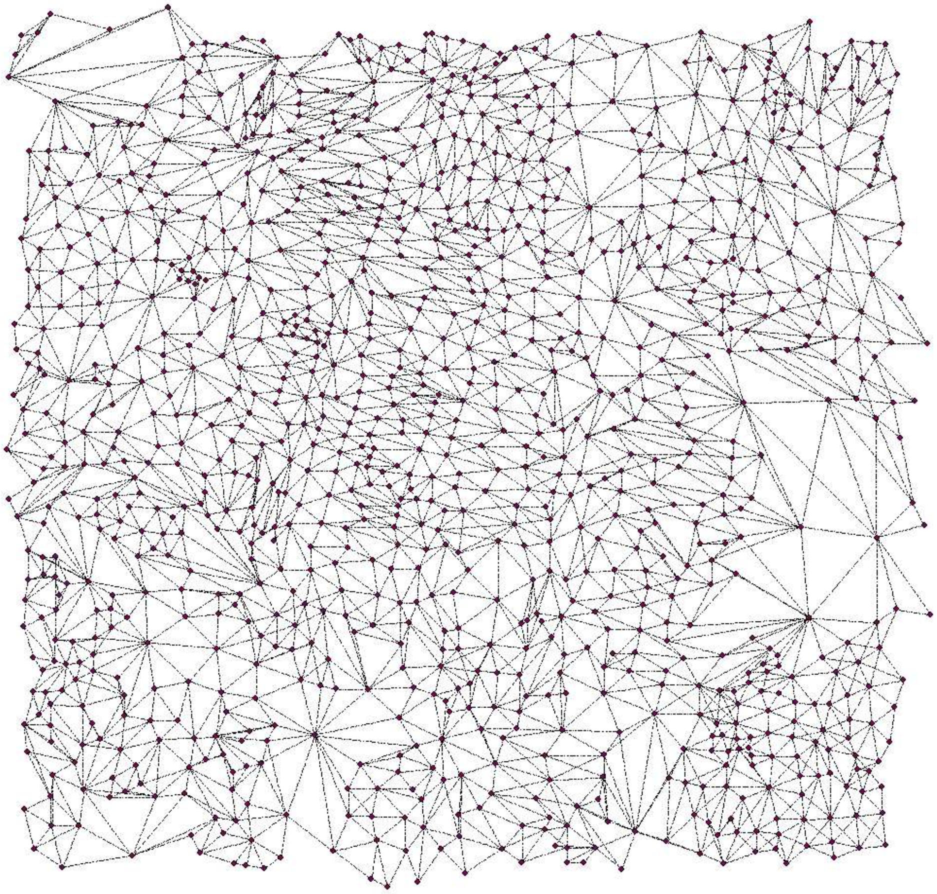
16

In terms of pseudo-code I am using these steps:

1. Define criteria for neighbours using spatial join 'one to many'. In this case it is the polygons that touch each other

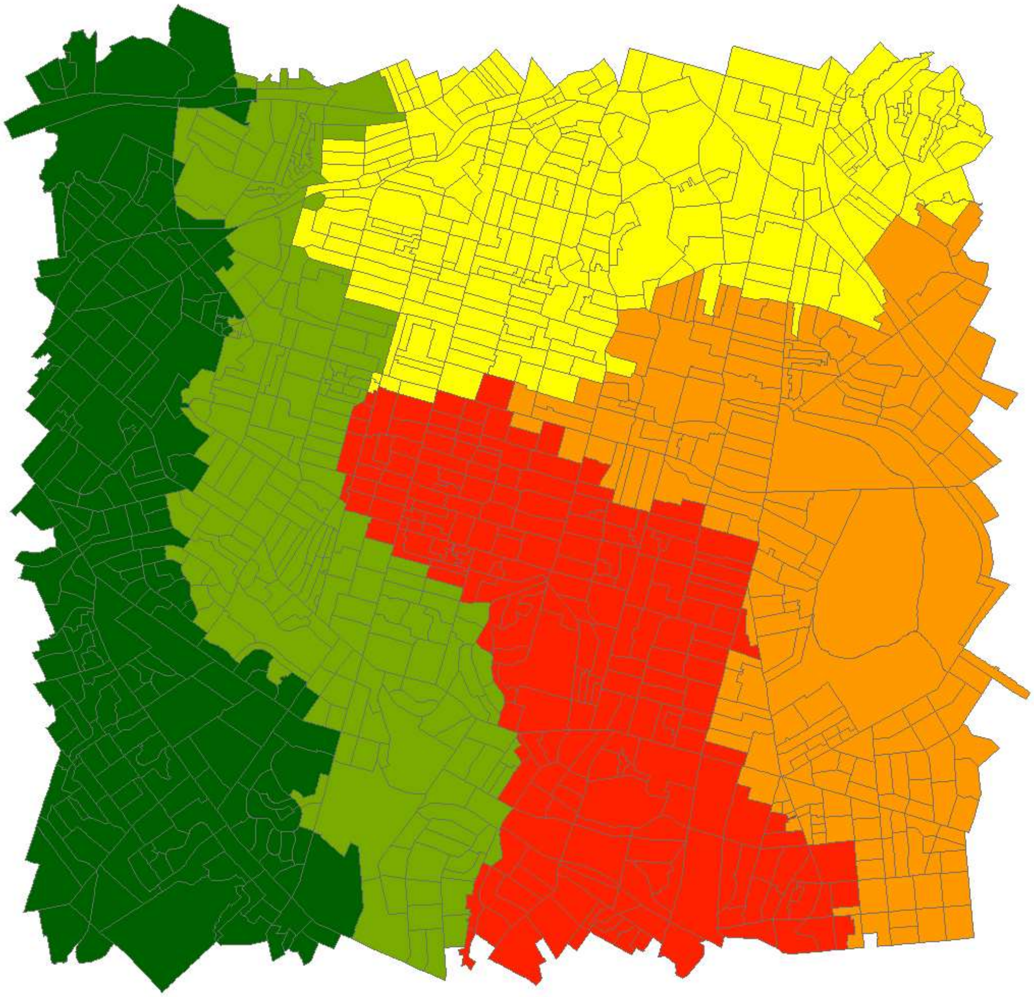


2. Connect neighbours by links

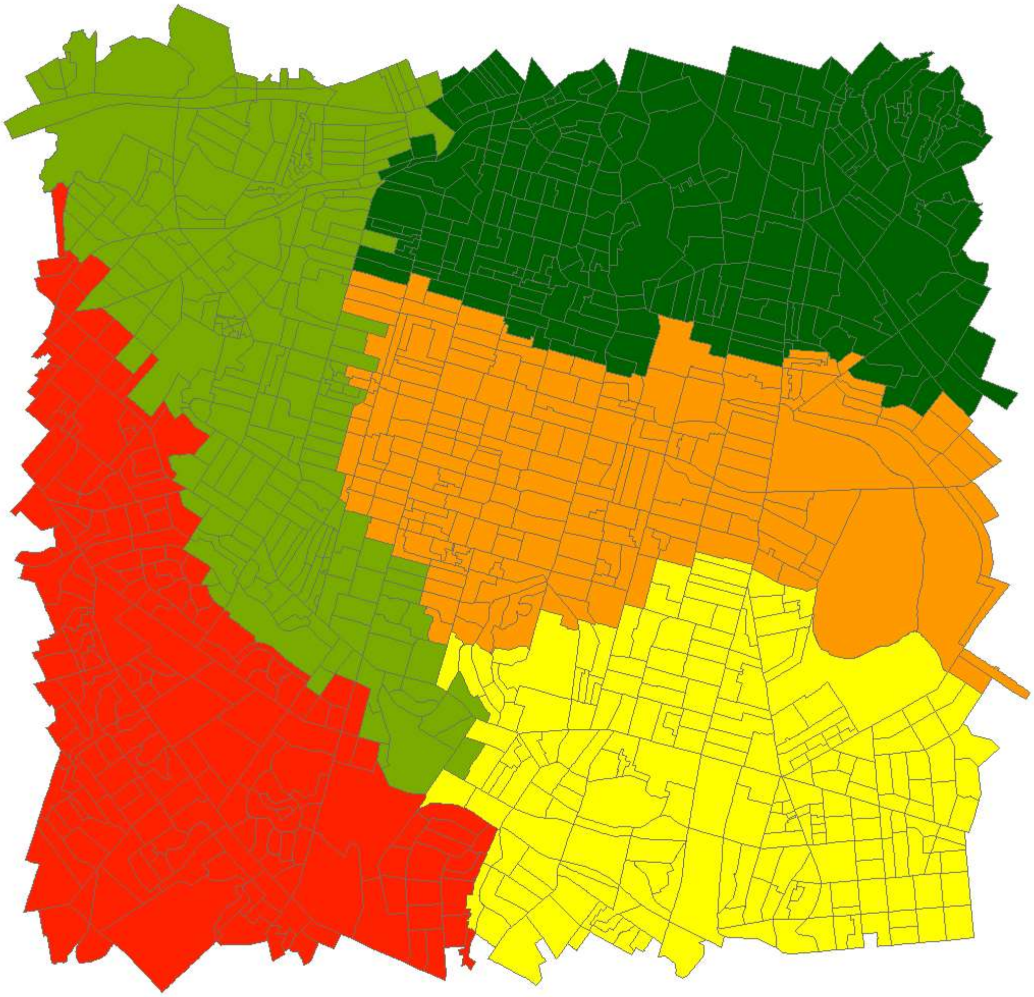


3. Use network analysis package, e.g. networkx to build a graph, with weights for all links =1.
4. Define a number of groups (N) and iterate through combination of nodes, e.g. p=itertools.combinations(aBig, 2), where aBig is a list of all graph nodes.
5. Calculate totals of population closest to 1st and 2nd node in a pair. Compute ratio. Break, if it is equal to (N-1) within tolerance of course. The group with a smaller population is your 1st group.
6. Reduce N by 1 and go to step 5 if N>0, otherwise stop

This show 1117 polygons split into 5 groups with approximately equal population



and this - with more or less the same area



Note that multiple solutions are possible depending on start node. In the examples shown it was in the lower left corner.

Share Improve this answer Follow

answered Nov 24, 2014 at 22:04

FelixIP
22k 3 27 57

2 That's an awesome answer to a vague question FelixIP! – Michael Stimson Nov 24, 2014 at 22:56

Thanks Michael Miles-Simpson. This 'grouping' puzzle was bugging me for ages. Was 'happy as' to find nx module lately and apply it to all sort of things and share it with ones who understand – FelixIP Nov 24, 2014 at 23:39

@FlexIP can you please share a sample code as I am beginner so can't get the pseudo code very well. – Shuji Feb 12, 2020 at 4:55

gis.stackexchange.com/questions/153094/... – FelixIP Mar 2, 2021 at 19:21