ASSIGNMENT 2

As stressed in Assignment 1, your approach to this assignment should be to write a *short report* on your findings for each of these three "studies". Since this is your first *analytical* assignment, it should also be stressed that each study should be a *pedagogical study* (as in the Example Assignment) that illustrates the application of the relevant analytical method(s). For example, when applying the *Clark-Evans Test* below, it is not enough to simply display the output from various software programs. You should give a short development of what this method tests, and exactly how it does so. What assumptions are involved? What test statistics are used? What are the advantages and possible limitations of this procedure? What information does it convey about the data set being studied?

(1) In this study you will focus on the type of point pattern analysis done in class for the "Redwood Seedling" data. Here you will use the "Iowa County Seat" data which is displayed in the ARCMAP file, **T:\ese502\arcview\projects\Iowa\Iowa_county_seats.mxd**.[1] Observe first that, unlike the Redwood Seedlings, this point pattern appears to be dispersed in a regular pattern (much like the "Cell Center" data discussed in class). If you open the 'Iowa counties' layer, you will see the actual county boundaries. In fact, the regularity of this grid-like pattern of county boundaries strongly contributes to the spatial regularity of county seats. But there is more to the story. Notice that especially for interior counties (away from the Missouri and Mississippi Rivers), each seat is near the center of its county. There are several references here that you should look at when interpreting this pattern. These references offer alternative explanations for landscapes like Iowa:

- The first is the classic discussion of "Central Place Theory" in the *Economics of Location* reference by August Lösch. Using the beer-producing area of Southern Germany as his key example, Lösch describes an idealized agricultural economy in which a hierarchy of towns, or "places", is regularly spaced on a hexagonal grid of small farms. [Pay particular attention to the highlighted sections]. This is actually quite similar to the case of Iowa, especially during the 1800s. (You can view Iowa as a "Löschian landscape" by closing the 'Iowa counties' layer, and opening the 'Voronoi_cells' layer above it.) A more general discussion of human settlement patterns is given in the *Locational Analysis* reference by Peter Haggett.

- The second pair of references are by Michael Dacey (*Dacey_1* and *Dacey_2*). Here Dacey offers an alternative probabilistic model of "place" patterns that are more regular than Poisson randomness. This is applied directly to county seats in Iowa (in both references). [Be sure to look my notes, *DACEY_model.pdf* (added to the back of Dacey_1) and also pay particular attention to the highlighted summary sections, *Pattern Summaries*, p.564 in Dacey_1, and *Evaluation and Interpretation of Results*, p.540 in Dacey_2.]

---

[1] This data is taken from the National Association of Counties website, http://www.naco.org.

With this background, we turn now to a point-pattern analysis of Iowa:

(a) First you will construct nearest-neighbors to each point using MATLAB:

1. Load the workspace **Iowa.mat** into MATLAB, and observe that the matrix, **L**, contains the locations of the 99 county seats (with coordinates in State Plane miles).

2. Construct the vector, **nn_dist**, of (first) nearest-neighbor distances in a manner similar to Problem 1(b) of Assignment 1. Use the commands:

   » **OUT = neighbors(L,1);**
   » **nn_dist = OUT(:,4);**
   » **nn_dist(1:2,:)**

   The last command is a check, and should yield the values 20.98 and 20.448. (For example, the nearest neighbor of county seat 1 is about 21 miles away.)

3. Now save this result to your own directory (say **S:\home**) by the command

   » **save 'S:\home\nn_dist.txt' nn_dist –ascii**

(b) Next, you will import this data to JMP and examine its properties.

1. Start by opening JMP. Reset preferences as in Problem 1(b).2 of Assignment 1 and import the file, **nn_dist.txt**, as a Text file. (Use "Best Guess" options and eliminate the empty "Column 1" if necessary.) Relabel the data column as "nn_dist". (Check to be sure the first two values are as above.) You will now use this to construct a file which is parallel to **Redwoods_data.jmp**, used for the analysis of the Redwood Seedlings in class.

2. Now add a second column, "rand_relabel", and use the formula: **Random →
   Col Shuffle**. After clicking **Apply** and **OK**, you should see a random relabeling of the 99 row numbers in this column.

3. Next, add a new column labeled "Sample", and as in Problem 1(b).3 of Assignment 1, construct the subscripted variable:

   $$nn\_dist_{rand\_relabel}$$

   This yields a random shuffling of the nearest-neighbor-distance values. The first 30 elements of this column then constitute the desired subsample of the nearest-neighbor-distance values to be tested.

4. Next, open the JMP file, **CE_Tests.jmp**, and copy-and-paste the first 30 values from "Sample" into the "nn_dist" column of this file.

5. To determine the appropriate area, open the ARCMAP file, **Iowa.mxd**, and observe from the attribute table of the layer, "Iowa_Boundary" that the last field contains the area of Iowa in square miles (56269 *sq.ml.*). This is the value to be used here. To enter this value into **CE_Tests.jmp**, right click on the "area" column, select **Formula**, and in the "Formula" box to the right, click on the blue box, type in the area value, and click **Apply** and **OK**.

6. Next, set the "n" value equal to 99 using the same procedure. (Remember that the *full* sample size is used to estimate point density.) The spreadsheet should now fill in the rest of the values.

7. Observe that the value "mu" gives the theoretical mean nearest-neighbor distance predicted by the *CSR theory*. Compare this value with the sample mean calculated in the column "s_mean". What can you conclude from this comparison?

(c) Finally, to carry out the **Clark-Evans test**, you can simply use the **P-value** columns of the table.

1. For a one-tailed test of "dispersion" versus the null hypothesis of CSR, the appropriate P-value is given by "P-Val Disp". Interpret this result.

2. How does this result compare with your findings in 1(b).7 above?

(d) Now you will repeat this analysis in MATLAB.

1. First load **Iowa.mat** into the workspace (found in **T:\ese502\matlab**). You will see one matrix, **L**, listing the locations of county seats. (*Note:* At this point you might also try the command **>> voronoi(L(:,1),L(:,2));** in MATLAB to see how it compares with ARCMAP).

2. Use the program **ce_test.m** (as done in class for **redwoods.mat**) with locations, **L**, area = 56269, and test = 2 (one-tailed test of dispersion):

   >> ce_test(L,56269,30,2);

   Compare the resulting **P-value** with that in 1(c).1 above.

3. Finally, use the program **ce_test_distr.m** (as in class) with number of tests, N = 1000.

   >> ce_test_distr(L,56269,30,2,1000);

   How do these results add to what you have already learned above?

Conclusions. In your concluding discussion of this study, consider how your findings relate to the Central Place Theory of Lösch. Also, how do they relate to the more statistical theory of Dacey?

(2) In this study (based on Exercise 3.3 in B&G), you will analyze the "Volcanoes in Uganda" data in the ARCMAP file, **T:\ese502\arcview\projects\Volcanoes\Volcanoes.mxd**. There is a key paper that you should look at before analyzing this data, namely the "Volcanoes in Uganda" paper by Tinkler in the Reference Materials. [Pay particular attention to the *Introduction* and *Conclusions* sections.]

- The present volcano data is taken from this study, and it should provide you with useful background material. Try to compare your results with his basic findings about clustering of volcanoes in the *Bunyaruguru field*. (Look at his map.)

- Pay particular attention to the notion of "areal volcanism" and how it relates to possible clustering of volcanoes.

- Also look at some of the additional PDF reference files that have been placed in **T:\ese502\arcview\projects\Volcanoes.**

(a) First you will export the point data from ARCMAP to MATLAB. To do so:

1. Open EXCEL (**Start > Excel** in Windows 10) and then open the data base (**.dbf**) file, **..\Volcanoes\Volcano_pts.dbf** (which is a component of the shape file for **Volcano_pts**).

2. In EXCEL delete the first row of the file (containing the column headings "X" and "Y") and save the file in your home directory as **volcano_pts.xlsx** using the save option: "Excel Workbook".

3. In MATLAB, under the **Home** tab, click on the icon, **Import Data**, and load the file **volcano_pts**. You will see the data appearing in the **Import** window. Be sure that both columns are selected, and for **Output Type** use the "Numeric Matrix" option (rather than the default "Table" option). You will see that the default title of the matrix is **volcanopts**. Rename to **volcano_pts** for clarity. Now on the far right, right click **Import selection** and select "Import data". The data matrix will then appear in the **Workspace** window.

4. Now (next to **Import Data**) click the icon, **Save Workspace**, and save the workspace in your home directory as **volcanoes.mat**. This is the workspace you will use for the remainder of the analysis.

5. To add the **area** of the Bunyaruguru field to this workspace:

    (i) Open the boundary file **..\Volcanoes\Uganda_bnd.dbf** in EXCEL and right click on the numerical "AREA" value to copy it. (This can also be done using the attribute table for **ugand_bd** in ARCMAP.)

    (j) In MATLAB type: **» area =** , and the paste in the value. (You could of course copy it by hand, but it is useful to be able to copy-and-paste sections of data.)

    (k) Now save the workspace again. (There is a *correct* version of this workspace, **volcanoes.mat**, in **T:\ese502\matlab**. So you may wish to compare your version with this workspace to be sure you have done everything correctly.)

(b) Next you will construct a version of the boundary file for the Bunyaruguru field in Uganda that is suitable for MATLAB analysis.

    1. First open **..\arcview\Volcanoes** again and copy the three (shape) files, **ugand_bd.shp**, **ugand_bd.shx**, **ugand_bd.dbf**, into your home directory. Be sure that this is in the same directory as **volcanoes.mat**, which I shall now refer to as **S:\Home**.

    2. To work with this shapefile in MATLAB, you will be using files in *both* your **S:\Home** directory and the class directory, **T:\ese502\matlab**. To do so, it is most convenient to reset the directory in the MATLAB window to your home directory, **S:\Home**, and to place the class directory on the MATLAB path.

    3. To do the latter, click on **Set Path** in the MATLAB main menu (below **Preferences**), and in the window that opens click on **Add Folder**. Now migrate to **T:\ese502\matlab** in the browser, click **Select**, and then click **Close** (if asked to save for future sessions, click **No**). Now all the programs in the class directory are available, and results in the workspace can be saved directly to your home directory just by clicking **Save Workspace**. (*Note*: This procedure might be useful for other Assignments as well.)

    4. To transform the shapefile, **ugand_bd**, into a boundary file suitable for MATLAB, you will use several MATLAB programs. The first program (part of the **spatial toolbox** in MATLAB), is called **shaperead**, and allows MATLAB to read shapefiles and import their relevant properties. The relevant command here is:

    >> [S,A] = shaperead('ugand_bd','UseGeoCoords', true);

    5. The option, **UseGeoCoords**, tells MATLAB to save the coordinate boundaries of all polygon features, which is what we need. This is saved in a MATLAB structure, **S**. We will discuss structures later, but for the present, if you click on

S in the workspace, you will see two vectors, **Lat** and **Lon**, containing the coordinates of the 396 boundary points in **ugand_bd**.

6. The next program, **shp_to_bnd**, (written by myself), reformats the data in **S** into a boundary file suitable for analysis in MATLAB (as illustrated on p.3-23 of Part I in the NOTEBOOK). The relevant command here is:

   >> ugand_bd = shp_to_bnd(S);

   This will produce an appropriate boundary file, **ugand_bd**, in the workspace. The first two lines should be "1 396" and "25740 41890".

(c) You are now ready to analyze this volcano point pattern in MATLAB.

1. First, plot out a **simulation envelope** for **99** random patterns by using the commands:

   » rand('seed',23456);
   » k_function_sim(volcano_pts,area,40,1,99,ugand_bd);

   The first line sets the random seed value (so that you all should get the same output).

   (i) Before proceeding further, save this figure to your home directory by clicking: **File > Save As** (in the Figure window) and saving the figure as **k_function.fig**.

   (j) What can you conclude from this graphical output? Do the results make sense when you compare them with the mapped point pattern?

   (k) If you would like to see some random distributions of volcanoes, write

   » poly = ugand_bd(2:end,:);
   » pt_in_poly_plot(120,poly);

   The first line creates a pure polygon by stripping the information row from the boundary file. The small program in the second line then generates 120 random points in **poly.** (Be sure the Figure window is closed so that you don't superimpose this plot with the envelope above.) If you want to print the results, export the figure to WORD, and use the "finishing" procedures discussed in (4) below.

2. Next construct a more detailed **P-value plot** for a new set of **99** random patterns. Here you will use the same random seed, **23456**, now specified in the information structure, **info**. This can be done with the pair of commands:

6

» info.seed = 23456;
» k_count_plot(volcano_pts,99,40,1,ugand_bd,info);

The use of the same random seed will ensure that the *same* data is generated to allow a better comparison of these two tests.

(i) Save this figure to your home directory as **k_count.fig**.

(j) Now (with **k_count.fig** still open) open **k_function.fig** for purposes of comparison, by clicking: **File > Open** (in the main MATLAB menu) and navigating to your home directory.

3. How do these results compare with those above? Is there any new information here?

4. For discussion purposes, you should include a print of these figures in your assignment report. A simple procedure is the following (using **k_count.fig** as an example):

(i) First click: **File > Export** (in the Figure window) and export the file to your home directory in **Enhanced Metafile (.emf)** format.

(j) Open WORD. [The following instructions are for **WORD 2016**]. click: **Insert > Pictures**, migrate to your home directory, and insert the file **k_count.emf** .

(k) Click on the image and resize it by dragging the corner with the mouse.

(l) To position the image with the image selected (so that the **Format** tab is open in WORD), select **Wrap Text > Behind Text**. You can then drag the image where you like.

(m) Finally, you can (if you like) add additional labels by using **text boxes**

  a. In the main menu select **Insert > Textbox**, and at the bottom of the window that opens, select **Draw Textbox**.
  b. If you left click on the spot where you want the label, then a **Textbox** will open.

  c. You can edit the **Textbox** by right clicking on the boundary of the box and selecting **Format Shape**.

  d. You can also use the mouse to drag the box and resize it (by again right click on the boundary).

Conclusions: In your concluding discussion of this study, consider how these results relate to the work of Tinkler. Are they consistent with his findings?

(3) In this study (based on Exercises 4.1 and 4.5 in B&G) you will analyze the "Myrtle Disease" data in the ARCMAP file: **T:\ese502\arcview\projects\Myrtles\Myrtles.mxd**. This data, which can also be found in the MATLAB workspace **myrtles.mat**, shows part of a grove of Myrtle trees in southern Tasmania that has been inflicted with a disease known as "Myrtle Wilt".

- The data set is taken from Jillian Packham's Ph.D. Dissertation (listed as *Myrtle Wilt* in the course Reference Materials). I have scanned portions of this work that will provide you with useful background material. [Pay particular attention to the highlighted portion of the discussion in *Section 2.5*.]

- Two shorter documents, *myrtle_wilt_dissertation.pdf* and *myrtle_wilt_explanation.pdf* are included in **T:\ese502\arcview\projects\Myrtles**. Additional material can be found by Googling "Myrtle Wilt", etc.

The object of this study is to analyze the spatial relations between diseased and healthy myrtle trees (diseased trees are shown in **red**). In particular, the task is to determine whether these spatial relations are consistent with contagion of Myrtle disease, or whether they are more consistent with random infections. A *contagion hypothesis* might be that the diseased trees appear in "clumps" or "patches". With respect to spatial dependencies *between* populations, this suggests that at scales where clumpiness is observed, healthy and diseased trees should show some degree of "repulsion". Similarly, this hypothesis also suggests that *within* the diseased-tree population there should be some degree of "clustering" at these scales.

(a) To address these issues, you will first use both the "random shift" and "random permutation (relabeling)" tests to examine possible spatial dependencies between diseased and healthy trees. The locations of diseased and healthy trees are given in the matrices **L1** and **L2**, respectively, and the dimensions of the box shown in **Myrtles.mxd** are given as (**xmin,xmax,ymin,ymax**). In the following steps, be sure to save and print out all plots from MATLAB as in Problem 2 above.

   1. Remember that (to avoid neighborhoods overlapping themselves on the torus) the maximum distance for shifts cannot exceed half the minimum of the width and height of the box. In this case,

   $$\min\{(xmax - xmin),(ymax - ymin)\}/2 = 46$$

   So the vector **D** contains a set of 11 representative distances between 1 meter and 45 meters to be used for the analysis. Here, you will use the **seed value**, 30456, to initialize the random number generator in the program. Hence, using the program **k12_shift_plot.m** for the *random shift test*, the command:

8

» k12_shift_plot(L1,L2,xmin,xmax,ymin,ymax,99,D, 30456);

generates a plot of the **P-values** for the distances in **D** based on 99 random shifts of the healthy trees (**L2**) relative to the diseased trees (**L1**).

2. What do these results tell you about the spatial relations between healthy and diseased trees? Are they consistent with the *contagion hypothesis* above?

3. Now try the *random permutation test* procedure, using the program **k12_perm_plot.m**. Here the relevant command is:

   » k12_perm_plot(L1,L2,99,D, 30456);

4. How do these new results differ from those in (2) above? Why might this be so? Do they add support to the *contagion hypothesis*?

5. Notice that in both the plots for **k12_shift_plot** and **k12_perm_plot** there appears to be some significant *attraction* between these populations at very small distances [D(2) = 3]. To check whether this makes sense, try the following commands in MATLAB (descriptions on the right):

» L = [L1;L2];	[stacks points to form single vector]
» n = size(L,1);	[counts number of rows in **L**]
» DIST = distance_mat(L);	[computes all pairwise distances for **L**]
» DD = DIST + 10*eye(n);	[replaces diagonal zeros with 10 (> 3)]
» D0 = DD(1:49,:);	[restricts **DD** to the **L1** rows]
» D3 = (D0 <= 3);	[binary form of **D0** with "1" for distances <= 3]
» rows = find(sum(D3′));	[finds rows with at least one distance <= 3]
» D3 = D3(rows,:);	[restricts rows of **D3** to **rows**]
» v1 = sum(D3′);	[counts all neighbors with distance <= 3]
» D3 = D3(:,50:end);	[restricts **D3** to the **L2** columns]
» v2 = sum(D3′);	[counts only **L2** neighbors with distance <= 3]
» MAT = [rows′,v1′,v2′]	[collects all results in a matrix]

   The last command (with *no semicolon*) then displays a matrix, **MAT**, in which the first column lists the row numbers of those *diseased* trees (**L1** trees) having at least one neighbor within a distance of 3 units. For each of these trees, the second column then shows exactly how many such neighbors there are. Finally, the last column shows how many of these are *healthy* trees (**L2** trees). If you have written these commands correctly, the first row of **MAT** should be [8 2 2], indicating that the diseased tree in row 8 of **L** has two neighbors within three units, and that both of these neighbors are healthy trees. [You can find each of these trees in ARCMAP by noting that FID numbers (which start at zero) are always one less than the corresponding row numbers in MATLAB.

So the tree in row 8 has FID number 7 in the Attribute Table for Diseased Trees.]

Based on the results in **MAT**, comment on whether the above significance result appears to be consistent with this information. (*Hint*: If the distribution were random, then (as a crude estimate) one would expect about 58.5% [= 100{69/(49+69)}] of these neighbors to be healthy trees. Compare this with the observed fraction).

(b) Finally, you will consider possible spatial relations *within* the diseased population (**L1**) relative to the healthy population (**L2**), again by means of "random permutations":

1. Using the program, **k2_diff_plot.m**, with the same data and a new **random seed**, 65846, the relevant command is now:

   » k2_diff_plot(loc,n1,99,D,65846);

2. Do these results lend support to the *contagion hypothesis*? Be specific in your answer.

   Conclusion: In your concluding discussion of this study, consider how your results relate to the findings of Packham (and the earlier work of Elliot that she discusses).