CHAPTER

SIX

VECTOR ANALYSIS

6.1 Geometric Measures

The term "geometric measurements" describes measurements on the spatial features themselves, and not on their attribute values. Geometric Measurements include:

- location.
- distance (distance between two features)
- length (length of a line segment or boundary of a polygon)
- area (of polygon features)

In this exercise, we will discuss all these measurements. A general overview of measurement operations on vector data is provided in your Living Textbook.

Important: Resources. You will require the latest LTR version of LTR version of QGIS, plus the dataset vector-analysis.zip. When you unzip the dataset, you will find the following files inside:

- `Vector_analysis.qgs` – a QGIS project preloaded with the datasets below;
  - `Centroids.gpkg`
  - `DistancePoints.gpkg`
  - `Linebuf.gpkg`
  - `Thiessenpoints.gpkg`
  - `Overlay1.gpkg`
  - `Overlay2.gpkg`

6.1.1 Location

The GIS always stores the location of the vector features. For point features the x and y coordinates are stored. For lines, the start node, end node and internal vertices are stored, and sometimes the length of the line segment. For polygons, it stores the line segments that define the boundaries, including the perimeter, and the area of the polygon. However, sometimes we also store the centroid of line or polygon features.

Task 1 In your dataset, you find a Shapefile called *'Centroids'*. Compute the centroids for the features in that layer in QGIS. Then, check if the centroids are inside or outside the original polygons. Fig. 6.1

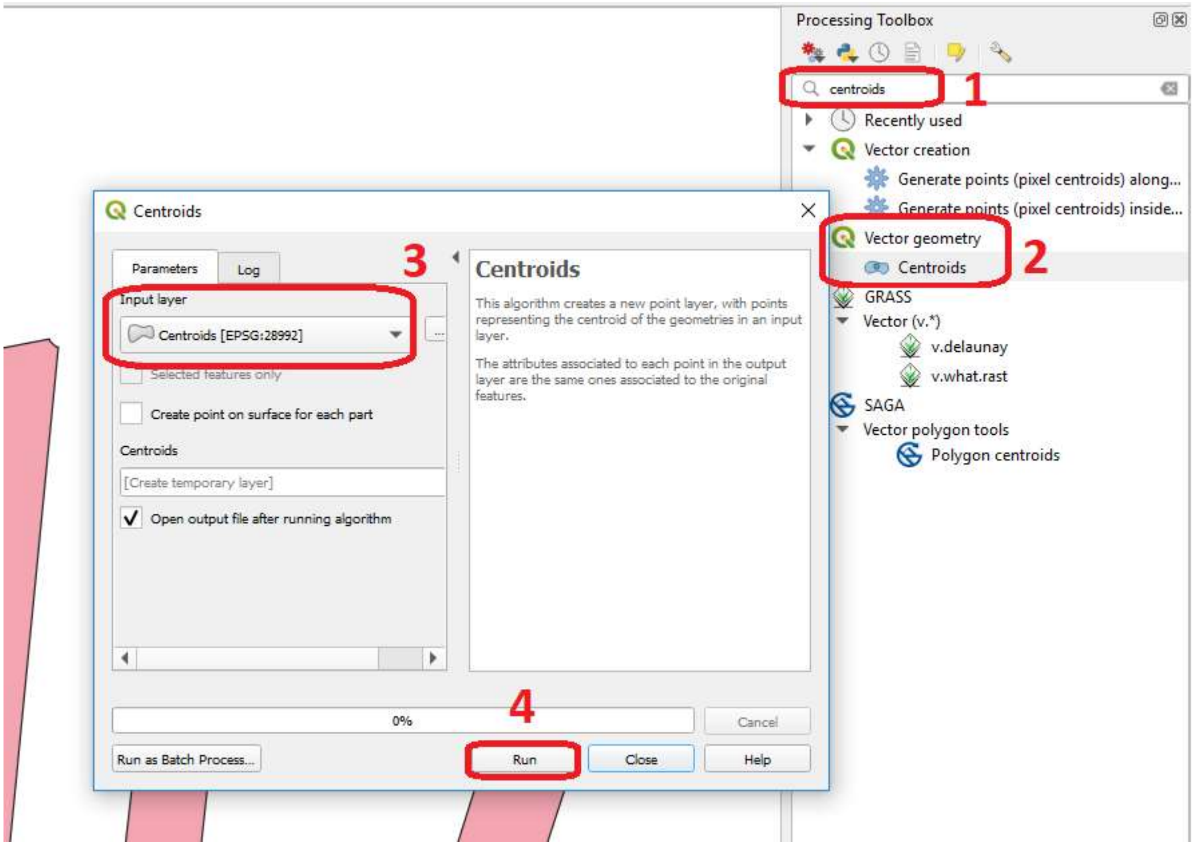


Fig. 6.1: Computing centroids in QGIS

Attention: Question. Can you give an example of a situation when computing the centroid is useful?

6.1.2 Distance

Another geometric measurement is distance. Computing the distance between two points in a straight line is a basic operation that you can solve using basic math.

Task 2 Open QGIS and use the **Add Geometry attributes** tool to find the exact coordinates of the poings in the *'DistancePoints'* layer. *Processing toolbox > Vector geometry > Add geometry attributes. The x and y coordinates will be added to the attribute table.*

Task 3 Using the x, y coordinates from the previous task, calculate **manually** the distance between the two points in meters. See Fig. 6.2

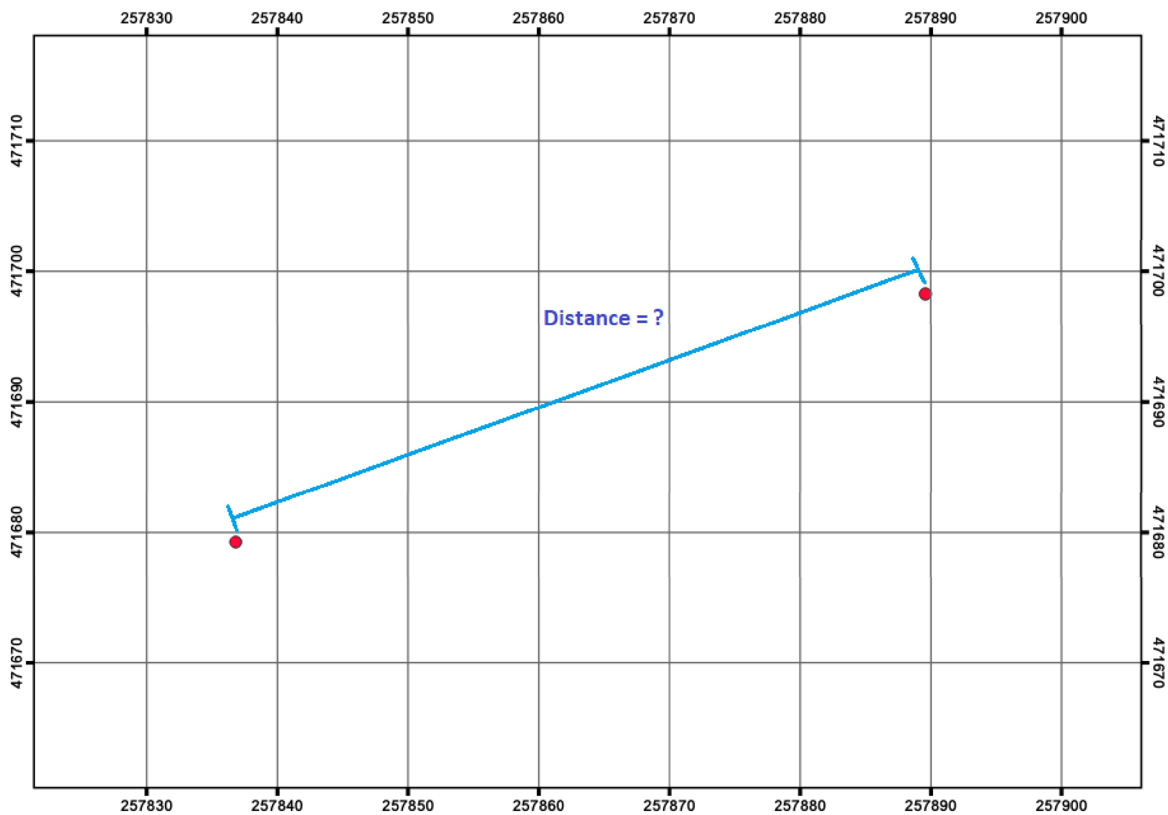


Fig. 6.2: Straight distance between points in the 'DistancePoints' layer

Task 4 Using the **Measure Line** tool, measure the distance between the points in the *'DistancePoints'* layer. Fig. 6.3

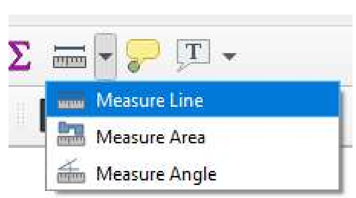


Fig. 6.3: Using the Measure Line Tool

Attention: Question.

- Measuring between two points is simple, especially when you use a measurement tool and draw the line you want to measure. However, in GIS software, some tools measure the distance from all features in one layer to the nearest feature in another layer. But, what would be the distance between a point and a line, or between a line and a polygon?
- The minimum distance between the features?
- The distance between the centroids of the features? or

- The distance between the two closest vertices?

Another type of geometric measurement discussed is the *minimal bounding box* of a feature.

Task 5 Use the **Bounding boxes** tool from the **Processing Toolbox** to visualise the minimal bounding boxes of the features of the *'overlay2'* layer.

6.2 Overlays

Vector Overlay operations combine two input layers (be it a point, line or polygon layers) into a new data layer. Vector overlay operations apply combinations of the following:

- Intersection of the geometry
- Spatial join of the attribute tables
- Definition of the output map extent

Some overlay operators perform both an intersection of the geometry and a spatial join of the attribute tables in combination with deriving a certain output extent. Still, others only join attribute tables or perform spatial intersections.

Task 6 Using the three polygons overlay operators discussed in the Living Textbook complete the table below.

Overlay Operator[1]	Intersection of the geometry?	Join attribute tables?	Output extent?
	(yes/no/partly)	(yes/no)	(AND/OR)

Task 7 Find the **Union, Intersect** and **Clip** tools in the **Processing Toolbox**, and use them to compute the overlay operations using the *'overlay1'* and *'overlay2'* layers as inputs. Compare the result with the table above.

"The fundamental operator of all these vector operations is **polygon intersection**. *All other operators can be defined in terms of polygon intersection, usually in combination with polygon selection and/or classification".* Below, you see the result of an overlay operation called: **Symmetrical Difference** between the *'overlay1'* and *'overlay2'* data layers. Fig. 6.4

[1] There are many other vector operators besides the operators discussed in the Living Textbook.

Fig. 6.4: Symmetrical difference bertween 'overlay1' and 'overlay2'

Attention: Question. How would you achieve the same results generated by the symmetrical difference tool, using only the *intersect tool* and *selection operators*?

6.3 Proximity Operators

We will cover two proximity operations: Buffer and Thiessen Polygons.

You create a buffer using point, line and polygon layers as inputs. Buffers can be created for all the features in a layer or for only a few selected features. We can use a **fixed buffer distance**; in which case, a buffer of the same size will be created for all the features in a data layer. However, we can also use a **variable buffer distance** for each feature; in which case such the buffer distances need to be stored in the attribute table of the layer.

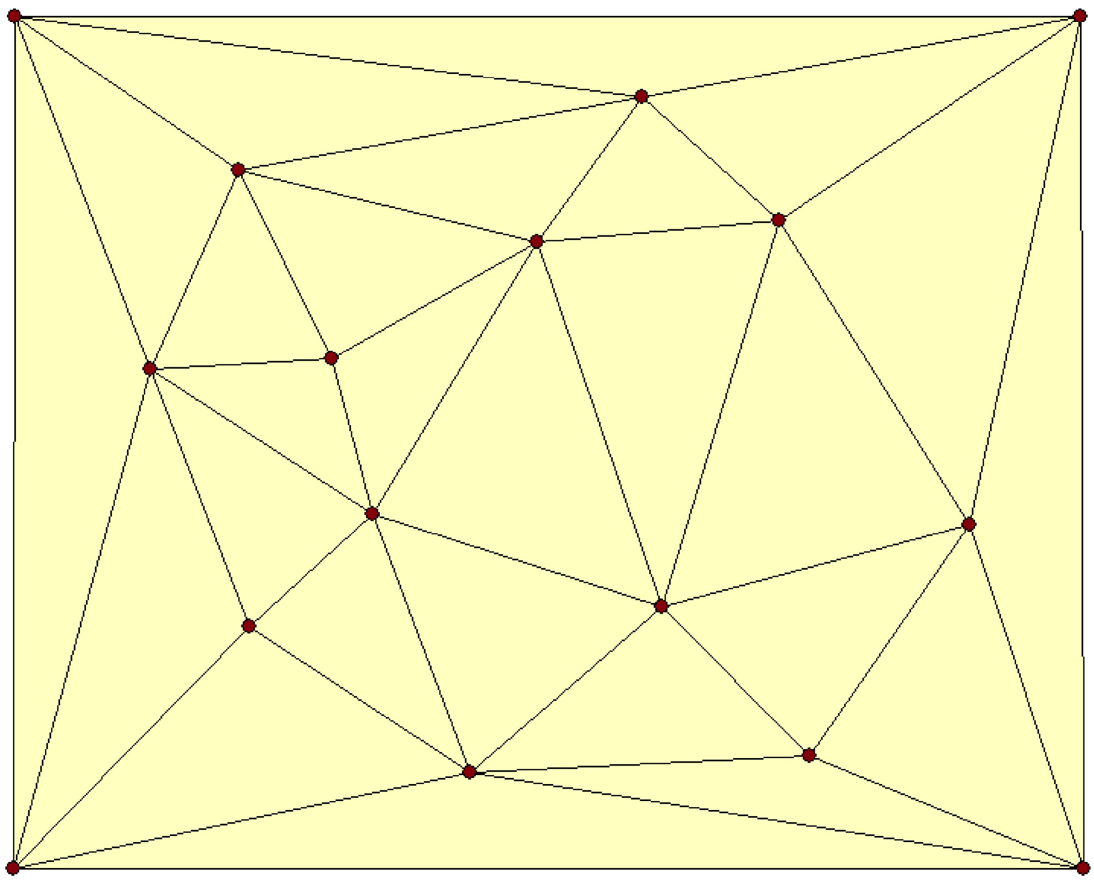
Task 8 Check the attribute table of the *'linebuf'* layer. You will find an attribute called **Bufdist**. Use this attribute to generate buffers with different buffer distances. Go to *Processing Toolbox* > *Variable distance buffer*.

Then, create a zonated buffer for the *'linebuf'* layer using a fix buffer distance. *Processing Toolbox* > *Multiring buffer (constant distance)*.

Attention: Question. One could argue that the problem with buffers is that they are discrete. Can you explain what that means and give an example in which that is a problem?

Another example of proximity operators is Thiessen Polygons. If you are familiar with the concept of *Voronoi Map*, Thiessen polygons are the same. They identify the areas that are closest (in *Euclidean distance*) to each point in a dataset.

Task 9 Below you see some points and a corresponding TIN (triangulated irregular network). Select 2 or 3 points and draw their corresponding Thiessen polygon.



Task 10 In the **Processing toolbox** search for a way to generate Thiessen polygons in QGIS. Remember that Thiessen polygons are also called Voronoi Maps and to find the correct tool in QGIS you might search for this term.

Note: Reflection. This website compares Thiessen Polygons with features in nature like the pattern on a giraffe: http://forum.woodenboat.com/showthread.php?112363-Voronoi-Diagrams-in-Nature

What do Thiessen polygons remind you of?