Craig Vis – 5873320

INFOMSDASM – Spatial Data Analysis and Simulation Modelling

Course Lecture Summary

Module 1.

*Lecture 1.a: Reference Systems*

*Lecture 1.b: Geodata Retrieval*

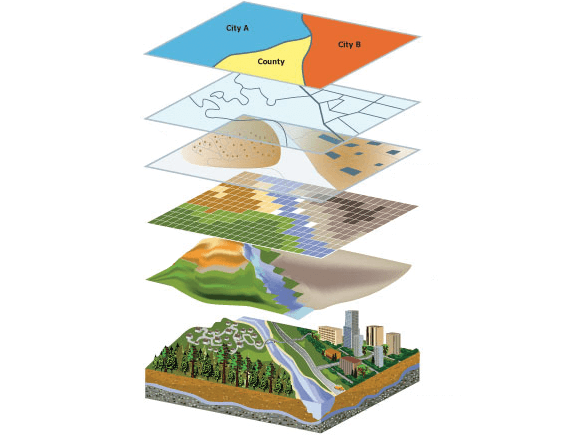
*Lecture 1.c: Geodata Quality*

Module 2.

*Lecture 2.a: Geodata Models and Core Concepts*

Principles of spatial data transformation:

* Layer principle:
  + Geographical information systems make use of layers of data to explore and study geographical and spatial phenomena.
  + Layers can be **‘overlaid’** to achieve several goals:
    - To spatially analyze landscapes
    - To derive new layers from already existing layers
    - To aggregate and summarize data of layers into new layers (Lecture 3.a)
  + Layers can either be **vector overlays** or **raster overlays**.
    - Vectors are values that are either points (individual values), lines (multiple values), and polygons (multidimensional values) that reflect real word features.
      * For example, you can place a height-layer on top of a soil-layer and on top of a vegetation-layer.
    - Raster function as individual cells equally spread out across a matrix, where each cell contains a value representing information (both quantitative (e.g., temperature as qualitative (e.g., land-use)). These cells are more than pixels, as they have cell coordinates and values. In addition, they can relate to both qualitative as quantitative characteristics. In Raster GIS, a **zone** refers to the set of cells sharing a certain value. A **region** is a zone with connected cells. Sometimes, cells might contain missing values. The cell size of a raster influences the **resolution,** which should be decided based on the needs of the analysis. A higher resolution requires more computation power.
      * For example, you can measure the suitability of a land-cell for development, by summing the cells for specific values appointed by a height-layer, soil-layer, and vegetation-layer.



^ An example of how layers of various geodata types are plotted to analyse an environment.

The question is: which GIS methods do you use to prepare and analyze a particular data source? Example questions are:

* When studying the amount of elderly in Amsterdam, do you make use of a **vector overlay method**, or **areal interpolation**?
* When studying CO2 pollution in Germany, do you make use of **aggregation** (I.e., summation) or **interpolation**?

Each method is used dependent on the data type that you are working with.

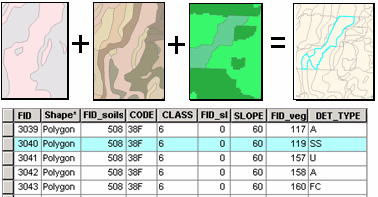
Core concept datatypes (CCD) ontology:

There are different models that work with these geometries, such as the *generalized geometric layer model*, the *vector geometry model*,and the **vector data model.** The latter is the model that we use the most, considering it is the most advanced and extensive. There exists an ontology that combines three dimensions to represent and study spatial data. These three dimensions are **layer types**, **core concepts,** and **levels of measurement**. The following section will study these three dimensions more extensively.

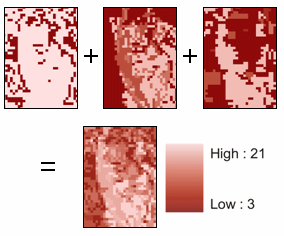
1. Layer types:

* **Region:** A dataset where the geometric primitives are regions in the forms of polygons or cells (e.g., parks in Amsterdam, or provinces in the Netherlands).
  + **Tessellation**: A subtype of region. A tessellation is a collection of plane figures that fills the plane with no overlaps and no gaps (e.g., Provinces in the Netherlands)
    - **Raster** (regular tessellation): A subtype of tessellation where the regions are cells in a matrix (e.g., height maps).
    - **Vector** (irregular tessellation): A subtype of tessellation that is not a raster (so the example of a map of Dutch provinces).
* **Lines**: Lines refer to coordinates that are extend on a map, that relate to networks (e.g. roads)
* **Point**: Points refer to individual coordinates on a map, that relate to particular objects that are placed on a map (e.g., trees)

In this *vector data model,* a data model consists of the before-mentioned **layers.** Each layer consists of particular **features,** which are **things** you can see, such as trees, houses, roads, and rivers. Each of these features contain their **geometry** and **attributes,** that contain the qualitative or quantitative information to describe the features.



^ An example of a vector overlay

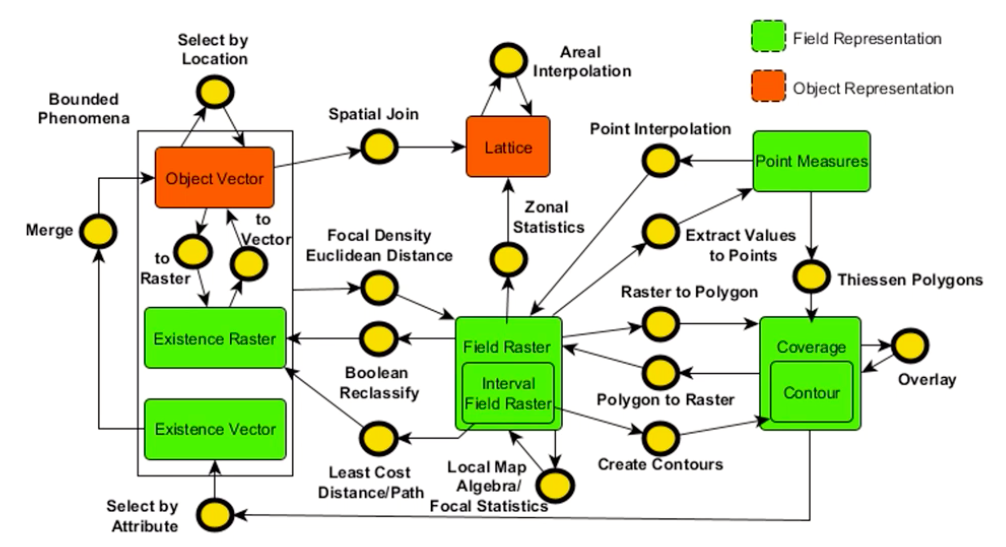


^ An example of a raster overlay

1. Core concepts:

**Core concepts** of spatial information relate to the way in which we can represent specific types of data. They are described as ‘cognitive lenses through which the environment can be studied’. These types of data can in some instances be **manipulated** to create other types of geodata. With regard to regions, *rasters* and *vectors* can often interchangeably represent the geodata as described below.

* **Field**: A field is a *continuous surface* of values in space.
  + **Contour**: A contour is a subtype of Field. Each contour encloses field values inside a given interval (e.g., a terrain contour map)
  + **Patch**: A patch is a subtype of Field. Each patch contains homogenous field values (so outside of a given interval) (e.g., dog-restricted areas in a park)
  + **Coverage:** Coverage is a subtype of Field. A coverage is similar to a patch, except that it is a tessellation, indicating that there are no overlapping nor gaps within the data. It is self-similar (I.e., values are consistent across other areas) (e.g., land use types)
  + **Field raster:** Field raster is a subtype of Field. Each cell in the raster represents the intensity of a value (e.g., Elevation map).
  + **Point measure:** Point measure is a subtype of Field. Point-like measurements are distributed across an area (e.g., temperature sensor measurements in a city).
  + **Line measure:** Line measure is a subtype of Field. Line measures represent a field in terms of lines of homogenous values (e.g., a coastline).
* **Object**: Objects are instances of *discrete values* that are named and are spatially bounded. Compared to Field, Object is a whole and indivisible entity of its own.
  + **Lattice**: A Lattice is a subtype of Object that represents not one point in space, but rather a tessellation. It is not self-similar (I.e., polygons) (e.g., values)
  + **Amount:** *NOT* an object on itself, and therefore not a subtype of Object. Instead, it rather represents several objects or matter.
* **Network**: A Network is a quantified relation between pairs of spatial objects. It relates to how object pairs measure against each other (e.g., traffic flow between two road intersections during rush hour).
* **Event:** An Event is a thing which happens in time and space. Events therefore always have a quantitative quality that is denoted as start and end time, as well as duration. The Event must be represented as a point in space, indicating where the event occurred. (e.g., earthquakes)
  + **Track:** A Track is a subtype of Event that are a collection of events demonstrating a trajectory of a moving object. Lines can be used to connect the tracking points. (e.g., a bird’s trajectory).

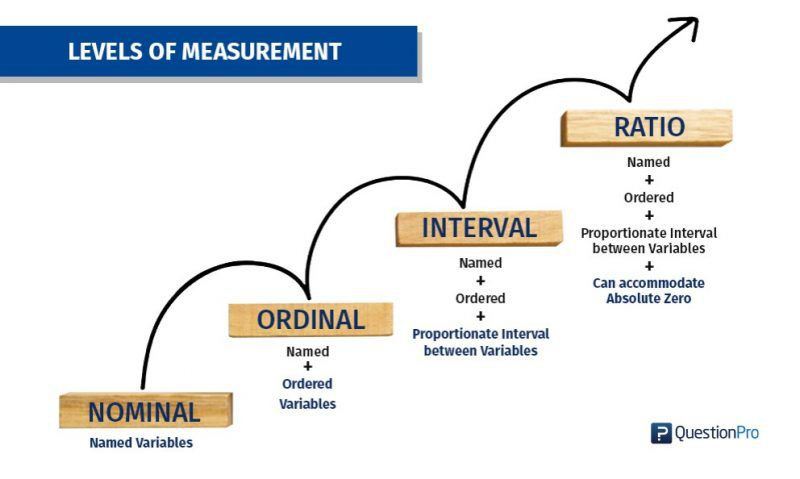


^ An overview showing the various manipulations that can be performed on geodata, demonstrating what input layer is needed, what manipulation is needed and what the consequent resulting output layer will become.

1. Levels of measurement:

Data can be of different kinds. The kind of data play a substantial role in the way in which they can or cannot be manipulated. Data generally is understood as:

* **Nominal**: Nominal data is arbitrary such as a name, or phone number, upon which no mathematical computations can be performed. They are unique identifier for values and are often used as primary keys.
* **Ordinal**: Ordinal data not only shows the identifier of a piece of data, but also shows how that data is ordered in relation to surrounding data.
* **Interval**: Interval data shows both the identifier and order between data, but also shows what the intervals, or gap are between these ordered data. Different interval levels can be interchangeable used while the data is preserved (e.g., temperature in K or C)
* **Ratio**: Whereas interval data shows the intervals between data points, it does not have a reference that indicates what the interval means. Ratio data have an absolute zero point, where the difference between numbers become significant.
* **Count**: Count data is a type of ratio data, where values are integer numbers representing a count of something that is whole.



^ A schematic overview showing the level of measurements. Each level is an advacement to the previous one.

*Lecture 2.b: Quality of Maps*

The quality of maps is dependent on multiple factors.

When visualizing data, not every **visual variable** (I.e., size, shape, color hue, color value) can be used on every data type. The level of measurement of the data influences how the data can be visualized. For instance, nominal values (I.e., **Discrete)** should not be visualized by means of **continuous** visual variables such as size or color value (that are intended for interval or ratio levels of data. Instead, for nominal values, use shape or arbitrary colors. Using continuous visual variables would suggest that there are either ordinal, interval, or rational differences between such nominal values, while this is not the case. On the other hand, continuous values should sensibly be visualized by attempting to project the continuity of the variable through the visuals. For instance, when using color on continuous values, you cannot simply use arbitrary colors to refer to a particular value, but instead opt for a spectrum of colors that allows for the transition to become apparent.

[The part within the presentation on ‘measurement scales’ has been added to the summary of the previous lecture, so that the ontology could be completed.]



^ A diagram showing which visual variables can be used on which data type.

Examples of maps are:

* **Choropleth map**. A thematic map where geographic areas are shaded or patterned in relation to a particular attribute value. These maps are **tessellated**. These values can be rational, from 0 to +1, but can also take on two colours, where the values range between –1 and +1. Do not use non-normalized
* **Proportional symbol map.** A map that uses a symbol at a specific or aggregate point, where the value is scaled by the size of a shape. Larger values are therefore larger symbols.
* **Dot density map.** A map that uses dots to indicate and represent data. The higher the density of the dots, the higher a value.
* **Contour map.** A map that uses differences in color hue (I.e., continuous scale) to demonstrate differences in value.

In his book on Thematic Cartography and Geographic Visualization, Slocum made a list of steps to go through when working on maps:

1. Is the map *general*, or *thematic*?
2. What is the spatial *dimensionality* of the map? (I.e., point, line, polygon, volume)
3. What is the *level of measurement*? (I.e., nominal, ordinal, interval, ratio)
4. Do the data require *normalization*? (I.e., is data absolute, or relative)
5. What is the *number of attributes* that need to be mapped?
6. What is the role of *time*?

*Lecture 2.c: Introduction to OpenStreetMap*

**OpenStreetMap (OSM)** is a collaborative project to create a free editable map of the world. The geodata underlying the map is considered the primary output of the project. People can attribute to the project voluntarily. It was partially started by dissatisfied cartography companies who had to pay large amounts of money for their products.

OSM data contains four elements:

* **Nodes**: Corresponds to a specific geographical point in space, containing the longitude and the latitude coordinates.
* **Ways**: An ordered list of between 2 to 2,000 nodes that define a polyline. These are used to represent linear features such as rivers and roads, but also buildings or forests.
* **Relations**: A multi-purpose data structure that documents a relationship between two or more data elements. They are used to explain how other elements work together.
* **Tags**: Descriptions about the meaning of the above-named elements.

OSM can easily be used in QGIS, by installing the plug-in and simply searching for nodes and tags.

*Lecture 2.d: Spatial Databases*

A **spatial database** is a database that is optimized for storing and querying data that represents objects defined in a geometric space. It is similar to a relational database but has some extensions that are used to store and manage spatial data like vector- and raster data.

Spatial data can also be stored in a **shapefile**, but because of the small maximum data storage of 2GB, spatial databases are usually preferred.

As mentioned above, multiple types of spatial data can be stored, like **vector data** (including points, lines, and polygons), and **raster data** (with differing resolutions based on the required analysis and outcomes). In addition, spatial databases can be used to perform various **computations**, like spatial join, spatial aggregation, and spatial computation. In addition, you can perform **spatial indexing** to facilitate the efficiency of queries.

One example of a spatial database, also the most popular one, is **PostGIS**. PostGIS is an extension of **PostgreSQL**, which is a free and open-source relational database management system. PostGIS adds spatial capabilities to the PostgreSQL relational database, which allows it to store, query and manipulate spatial data.

PostGIS defines various spatial datatypes:

* **Geometry**: Represents a feature in planar (Euclidean, I.e., x-y) coordinate systems. The geometric types that are recognized by PostGIS are points, multipoints, lines, polygons, and more.
* **Geography**: Represent a feature in geodetic coordinate systems. Geodetic coordinates are spherical coordinates expressed in angular units (degrees).
* **Raster**: Represents raster data (imported from TIFFs, PNGs).

Like QGIS, a **Spatial Reference Identifier (SRID)** is needed to associate the coordinates with a specific coordinate system. In the Netherlands, we use SRID:28992 --- Amersfoort / RD New. The World Geodetic System is SRID:4326 --- WGS 84.

PostGIS recognizes several spatial functions, that can manipulate geographical data:

* Creation of points, lines and polygons: ST\_MakePoint, ST\_MakeLine, ST\_MakePolygon
* Access to geometries: ST\_StartPoint, ST\_EndPoint, ST\_X, ST\_Y
* Access to properties: ST\_IsValid, ST\_IsClosed, ST\_Npoints, ST\_IsSImple
* Edit geometries: ST\_AddPoint, ST\_Multi, ST\_Translate
* Output geometries: ST\_AsText, ST\_AsKML, ST\_AsGML

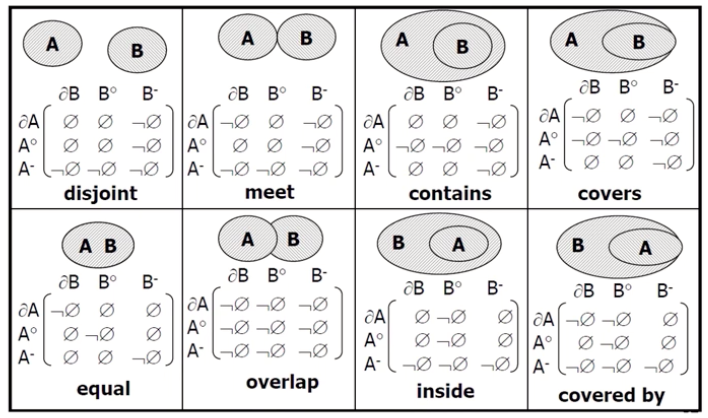
The lecture elaborates on the various functions, and how they can be used to create, or manipulate the data.

Module 3.

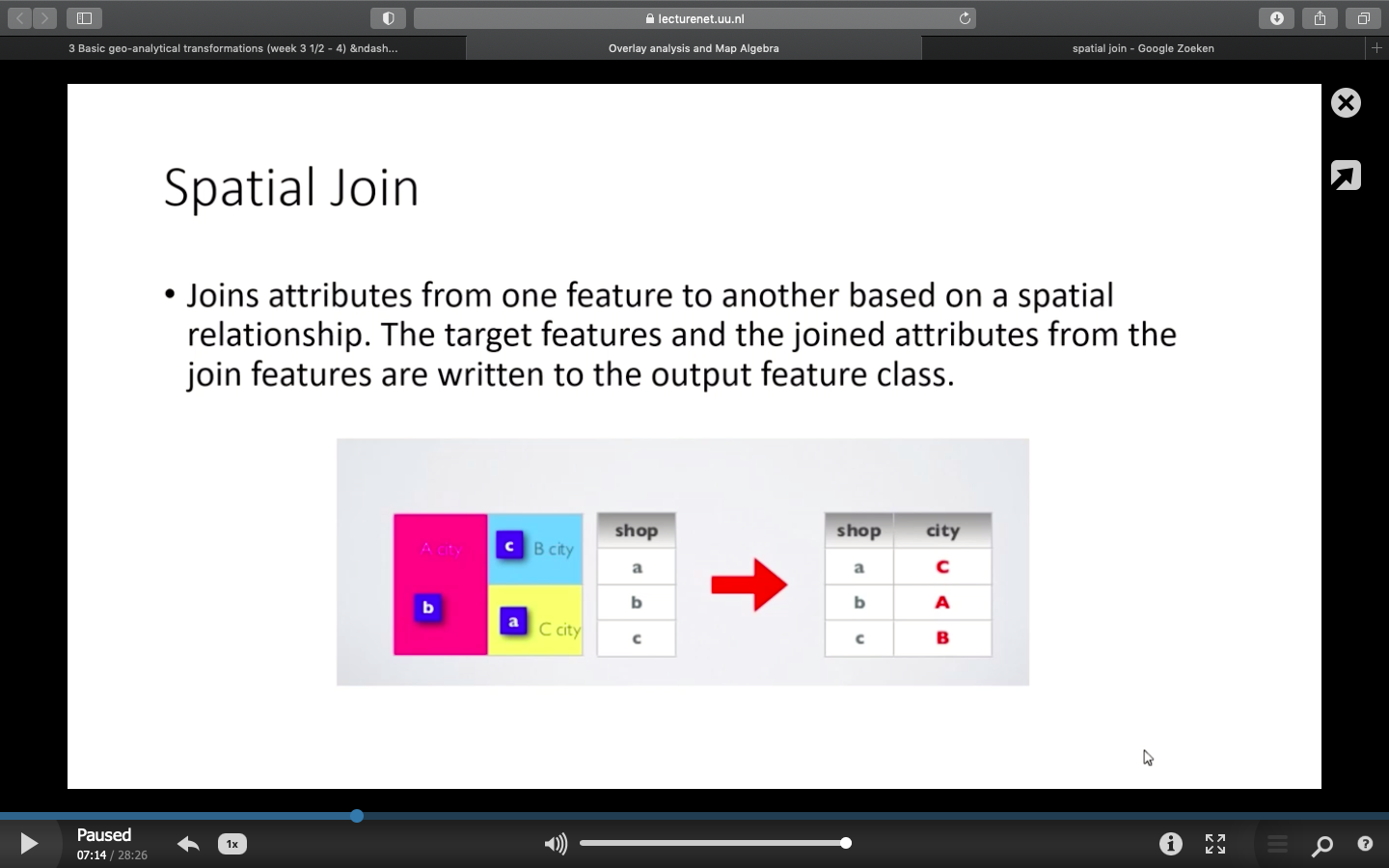
*Lecture 3.a: Overlay Analysis*

In Lecture 2.1, we discussed how layers can be visualized on top of each other. By doing so, you can perform various overlay analyses on both vector overlays and raster overlays. For each specific overlay, you perform different computations.

* Vector overlay:
  + **Attribute Relations:**  Like relational databases, spatial data is stored in **rows** (I.e., entities) and **columns** (i.e., attributes). In QGIS, you can use the **Field Calculator** to add, edit or remove fields. To study uniquely identifiable rows, one of the attributes must be a **primary key**. We call this an **attribute relation**. You can join relations by merging the data of both relations on the primary key.
    - In QGIS, an Attribute Join can be done by using the **Join Attributes** function in.
  + **Spatial Relations:** In addition to these attribute relations, spatial databases can make use of **spatial relations** to join relations, or in other words, construct new polygons**.** Such manipulations therefore use the values (I.e., boundaries and inner coordinates) of the two layers of polygons to compute a new entity. **Spatial Join** joins attributes from one feature to another based on a spatial relationship. The target features and the joined attributes from the join features are written to the output feature class.
    - In QGIS, a Spatial Join can be done by using the **Join Attributes by Location.**

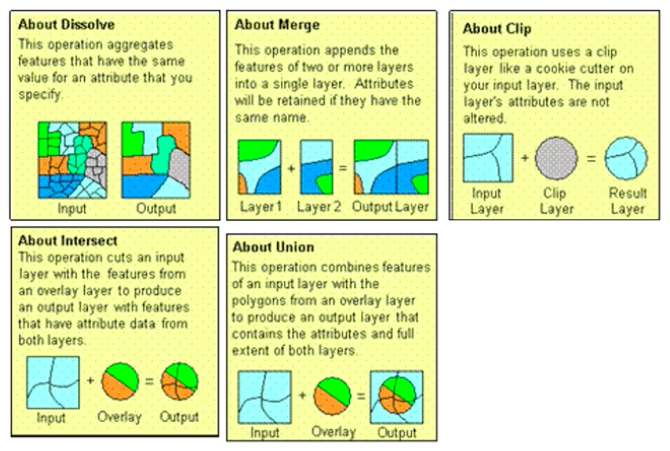


^ An example of the various computations that can be used to establish new relations that create new layers.



^ An example of **spatial join**, where two overlapping layers, one being a point layer, and the other being a field layer, are written to a new feature where the two layers are joint.

* + **Other overlay operations:** Additional operations can be applied to layers that either dissolve (or aggregate), merge, clip, intersect or union the layers. Here, the inputs cannot be points, but must be **lines** or **polygons**. From this, new lines or polygons are outputted.
    - In QGIS, these operations can be used by selecting the appropriately named functions (I.e., **Dissolve, Merge, Clip, Intersect and Union)**

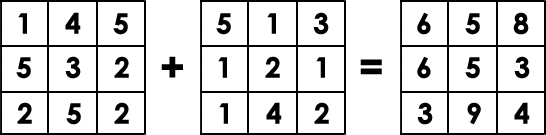
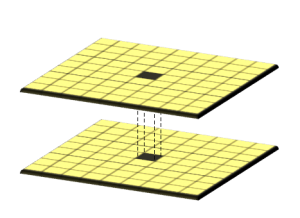


^ Examples of the other **overlay** operations dissolve, merge, clip, intersect and union to define new layers.

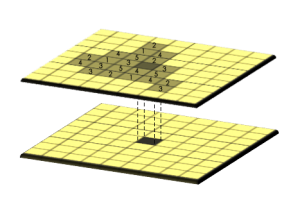
* + **Areal Interpolation:** While this course uses a generally simplified approach of interpolation, it can be understood as an area weighted average. There are more sophisticated techniques to do so which will be discussed in the lecture on Distance-Based Analysis.
* Raster overlay: (<https://gisgeography.com/map-algebra-global-zonal-focal-local/>)
  + **Map algebra**: A **cell-by-cell** combination of raster data layers using algebraic operations. These are simple operations that are performed on the numbers that are stored as values within the raster cell locations. The basic elements of map algebra are:
    - **Objects** (I.e., datasets, layers, and values (as input or storage)
    - **Operators** (I.e., addition, subtraction, multiplication, etc.)
    - **Functions** (I.e., loc, foc, zon, glob)
    - **Actions** (I.e., the results of applying such functions)
      * **Qualifiers** on the actions decide the functionality of the action.

Such functions are said to be **higher-order functions** as they take an algebraic function and apply it to a set of cells. This set of cells can be **local** (I.e., cell-by-cell), **neighborhood**/**focal** (I.e., moving neighborhood), **zonal** (I.e., within a homogenous zone), **global** (I.e., the full dataset).

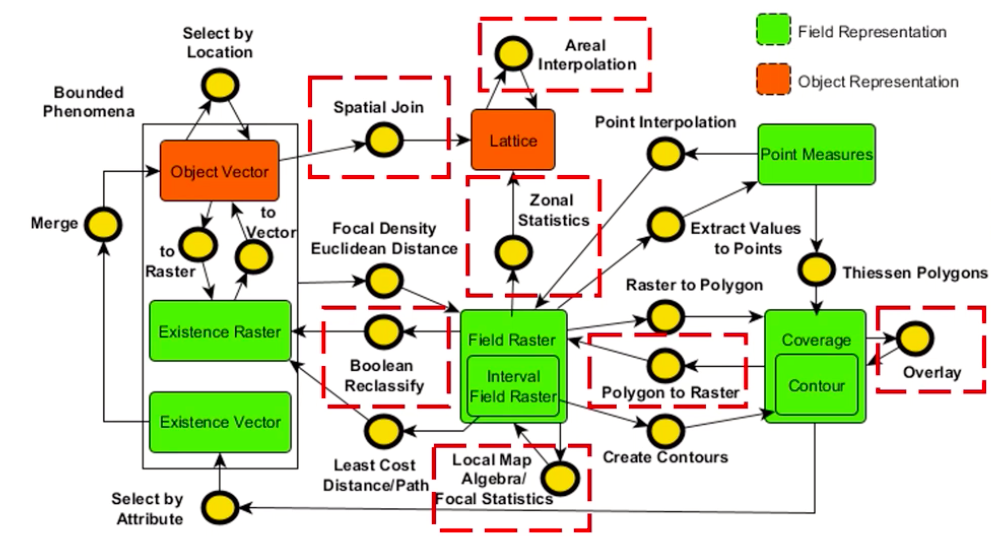
* + - In QGIS, use the **Rasterize (vector to raster) (I.e. Polygon to Raster)** function to perform convert vectors to rasters. You select the attribute in the input vector that should be written into the cell values of the raster.
    - In QGIS, use the **Reclassify by table (I.e. Boolean reclassify)** function to turn a raster in a new raster where each cell value is changed to a new value based on values from another table.
    - In QGIS, use the **Raster Calculator** to write down map algebra, where you can perform the above-named operations.
    - In QGIS, use the **Zonal Statistics** to take the vector polygon layers that defines the zone, and the raster layer, to compute a new raster layer that uses the zone of the vector polygon layer to calculate new values for the cells that lie in those zones. In QGIS, use the **Processing Modeler** to save/ rerun processing workflows for raster and export them into Python.



^ An example showing local map algebra.



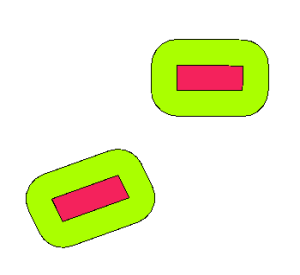
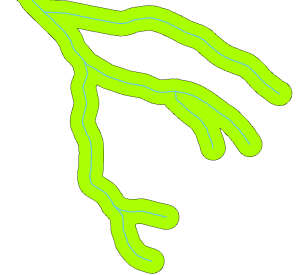
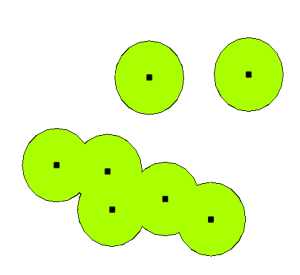
^ An example showing zonal map algebra. (Neighborhood & global next lecture)



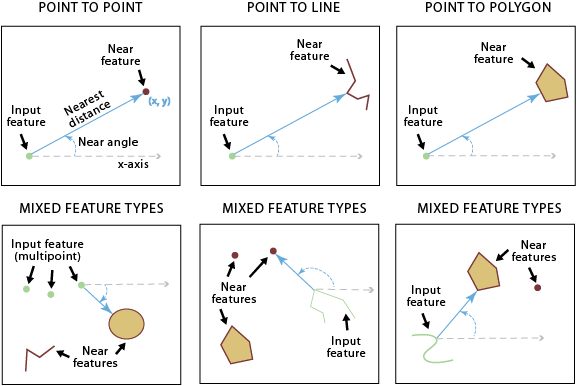
^ These are the manipulations that we have performed in this lecture. The manipulations are highlighted as red in the summary above.

*Lecture 3.b: Distance-Based Analysis*

* Vector Distance-Based Analysis
  + **Planar** distances are measured in a **projected coordinate system (CRS)**. These distances, however, are **distorted** on the spherical earth surface, or in other words, the **geodesic** distance. Therefore, one needs to account for varying distance measurements over the different planar and spheroid models. As QGIS does not support measurement on spheroid models of the Earth, but only Cartesian models instead, you need to take this into account when calculating these distances.
  + Three distance-based methods/ proximity analyses include:
    1. **Buffers**: A buffer creates two areas: one area that is within a specified distance to the selected real-world features, and one area that is beyond the selected real-world features. The area within the specified distance is called the **buffer zone**. A *buffer zone* is any area that serves the purpose of keeping real world features distant from each other (e.g., greenbelts between residential and commercial areas, border zones between countries). These *buffer zones* are vector polygons that enclose the *buffer points, lines, or polygons*. The **buffer distance** or buffer size **can vary** according to numerical values provided in the vector layer attribute table for each feature. The numerical values must be defined in map units according to the Coordinate Reference System (CRS) used with the data. For example, the width of a buffer zone along the banks of a river can vary depending on the intensity of the adjacent land use. For intensive cultivation the buffer distance may be bigger than for organic farming.Buffers can be **merged/ dissolved** into a single geometric object to avoid overlapping areas, which is recommended to minimize the number of geometric objects.
    - In QGIS, use the **Buffer** function to perform Buffer operations onto the data.

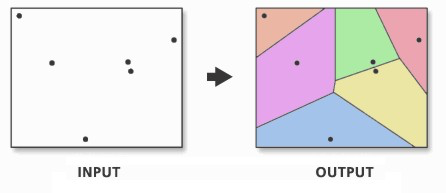


* + 1. **Nearest objects:** The ‘distance to nearest’, ‘nearest object’, or ‘least cost/ distance path’ calculates the distance and additional proximity between the source/ input feature and the closest feature in **another layer**. This can be done for points, lines, and polygons.
    - In QGIS, use the **Distance to Nearest Hub** function to calculate these distances.



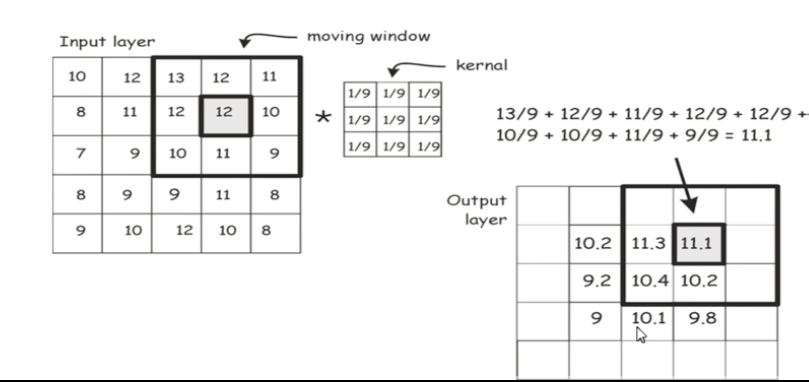
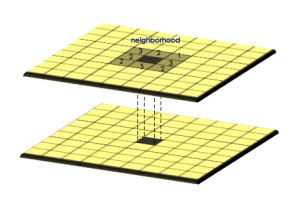
^ Three proximity calculations from point to point, point to line, and point to polygon.

* + 1. **Thiessen Polygons:** Sometimes also known as Voronoi Diagrims/ Dirichlet tessellations, these polygons exactly bisect distances between **points** using a Delaunay Triangulation. A polygon is constructed around the point, splitting the area up into **polygons**, rather than **points**. So, the **points** within a polygon are those points that lie the closest to its particular center point, rather than to points outside of this polygon. This can be used to construct catchment areas or to interpolate measures.
* In QGIS, **Thiessen polygons** can be made using the **Voronoi Polygons** function.



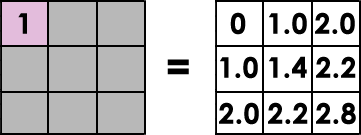
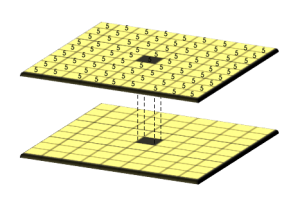
^ An example of how Thiessen polygons are constructed around points, where the values within a polygon lie the closest to the center point within that polygon.

* Raster Distance-Based Analysis: Whereas local and zonal maps were discussed in the previous lecture, the distance now increase. As a result, different computations are required for these map algebras.
  + **Focal/neighborhood map algebra:** The f**ocal/neighboorhood function** (foc) is performed on cells within a **moving cell neighborhood**. A moving window is a rectangular arrangement of cells that **shifts in position**. By applying an operation to **each cell** from a moving window, it commonly smooths values in a raster. Focal functions can be of different shapes. A **kernel** is a set of constants for with the values within a window are multiplied. Kernels are used to find concentration or elevation changes.



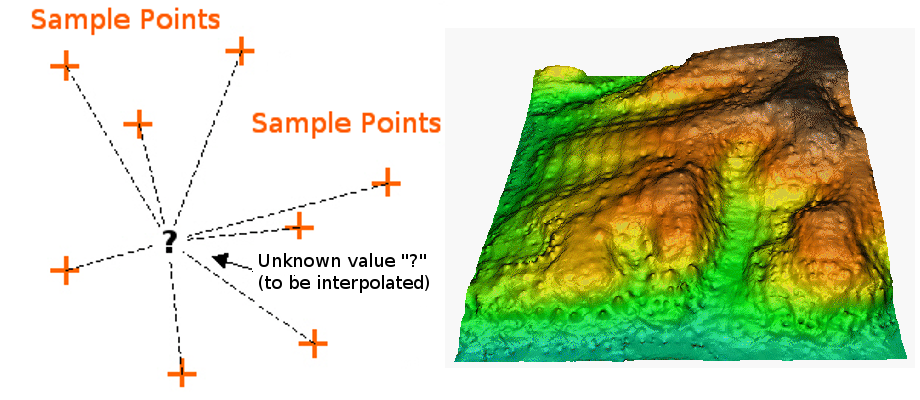
^ An example showing focal map algebra. ^ An example of a moving window and a kernel

* + **Global map algebra:** Global operations (glob) apply a bulk change to **all cells** in a raster. If you want to add a value of 1 to all grid cells, this is a global operation. For example, Euclidean distance is an example of a global operation. By calculating the closest distance away from a source, it applies the function globally in a raster.

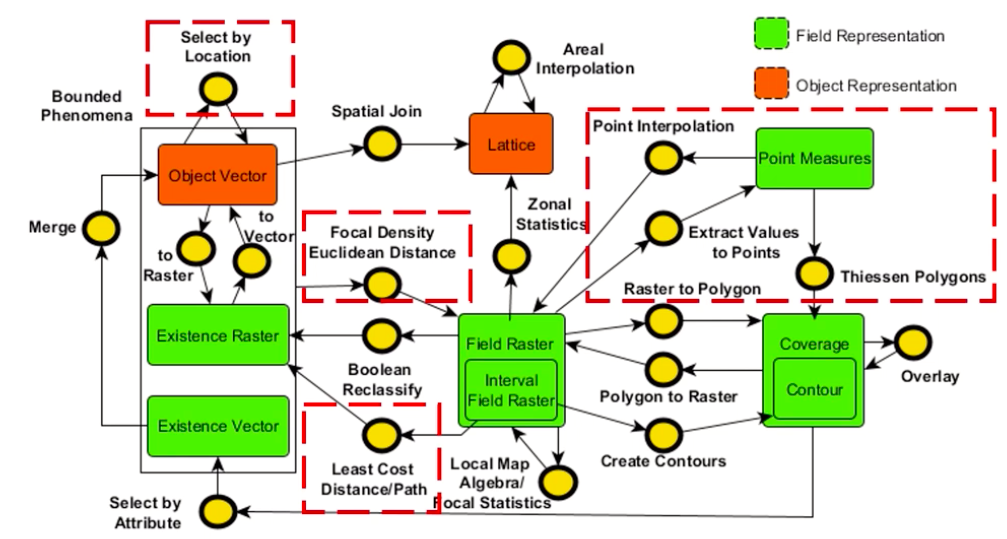


^ An example showing global map algebra. Values increase with distance of the source raster.

* In QGIS, you can calculate the proximity by using the **Proximity (Raster Distance)** function.
* In QGIS, you can calculate the focal statistics using the **r.neighbors** function.
* In QGIS, you can calculate the kernel density by using the **Heatmap (Kernel Density Estimation).** On the map, it is called **Focal Density**.
  + **Point interpolation:** Interpolation **predicts** values for cells in a **raster** from a limited number of **sample data** points. It can be used to predict unknown values for any geographic point data, such as **elevation**, rainfall, chemical concentrations, and noise levels. **Inverse Distance Weighting (IDW)** uses the distance, where an increasing distance causes the factor to diminish. Closer points weigh heavier than more distant points. A **Kriging** is more sophisticated than IDW, giving a smoother picture, and adds a standard error.In IDW, you can change the parameters.
    - The lower the exponent n, the more distant points affect the interpolation. The greater exponent n, the more local points affect the interpolation.
    - The search neighborhood can be constrained to for instance 5nn, 12nn, and 25nn. The lower the nn, the less sharp the image as the averaging factor will be less.
    - In QGIS, you can apply **IDW** by using the **Grid (IDW With Nearest Neighbor Searching Function)**.



^ An example of point interpolation.



^ These are the manipulations that we have performed in this lecture. The manipulations are highlighted as red in the summary above

*Lecture 3.c: Spatial Network Analysis*