Personal understandings

One of the difficulties of this project is the understanding of various concepts. Because of translation problems, people have a lot of confusion about the definition of technical terms and how to calculate them. The following are some of my personal understandings

DEM

DEM (Digital Elevation Model) data is a 2D raster dataset that expresses surface elevation in raster form. DEM data can be viewed as a raster grid where each pixel represents the elevation of a specific point

Gradient and slope

Gradient is usually a vector value representing the direction and magnitude of the height change. Gradient can be obtained in GIS by calculating the height difference between adjacent pixels in the DEM dataset. Gradient calculations can be used for many terrain-related analyses, such as flow direction, viewshed, and terrain curvature.

In GIS, when calculating the slope, the height difference ratio refers to the ratio of the height difference between two adjacent pixels to the pixel spacing. Slope is usually calculated using the following formula:
slope = arctan(height difference/pixel pitch)
Among them, arctan is the arc tangent function, the height difference is the height difference between adjacent pixels, and the pixel pitch is the horizontal distance between adjacent pixels. The slope calculated by this formula is usually expressed in degrees or percentage, where degrees are expressed as angles, and percentages are expressed as the ratio of height difference to pixel pitch multiplied by 100.
For example, if the height difference between adjacent pixels is 5 meters and the pixel spacing is 10 meters, the slope is arctan(5/10) = 26.6 degrees or 50%.

Aspect

In GIS, aspect refers to the orientation of the terrain surface on the horizontal plane, that is, the maximum slope of the ground in a certain direction. Aspect can be expressed in angle or azimuth, usually in degrees.

The method of calculating aspect is usually based on DEM dataset. Aspect can be obtained by calculating the height difference and direction between the surrounding pixels of each pixel. Aspect is usually calculated using the following formula:

aspect = arctan((dY/dX))

where dY and dX are the height difference and horizontal distance between adjacent pixels in the DEM dataset. arctan is the arc tangent function, and the return value is radians, which need to be converted to degrees.

Aspect is usually expressed from 0 to 360 degrees, where 0 degrees is due north, 90 degrees is due east, 180 degrees is due south, and 270 degrees is due west.

Comprehension of displayed images

In data visualization, a colormap is a way of mapping values to colors. The choice of color can reflect the characteristics of the data, making it easier to identify patterns and anomalies. Both "viridis" and "hsv" are commonly used colormaps, "viridis" is a gradient colormap with different colors, where lower values are shown as blue, higher values are shown as yellow, intermediate values are shown as green, and " hsv" is a colormap of Hue, Saturation, Value (Hue, Saturation, Value) mode, which can display angle change information.

Interpolation is the process of using function values from known data to estimate function values at unknown points. The "nearest" interpolation method refers to finding the nearest data point around a discrete data point and using the value at that point to estimate the value of a function at that point. In data visualization, this method can make the image clearer and avoid the blurring phenomenon when there are too many pixels.

In this assignment, the imshow() function is used to display the maximum slope data and slope direction data, where the colormap is "viridis" and "hsv" respectively, and the interpolation method is set to "nearest". This can make the image clearer, easier to observe and analyze, and also in line with the basic principles of data visualization.

Reflection

Why elevations above sea level

In elevation data, height values below sea level can affect the comparison and analysis of terrain heights, causing changes in height values to appear larger than they actually are. This is because the origin of elevation data is usually sea level, and changes in terrain height are calculated relative to this datum. If the data contains height values below sea level, these values cause the datum to be lower than sea level, making terrain heights appear higher relative to this datum.

In this example, if the elevation data contained height values below sea level, those values would cause the datum to be lower than sea level, making the overall terrain height appear higher relative to this datum. So, in this example, the peak heights of the mountains appear higher, while the lowest point heights of the valleys appear lower, because their heights have increased relative to this datum. This can be misleading for comparison and analysis of terrain heights, leading to an inaccurate understanding of changes in terrain heights.

Therefore, when analyzing elevation data, it is usually necessary to emphasize that the height values in the dataset are based on sea level to ensure the accuracy of comparison and analysis of height values.

If the given data has data below sea level, consider using relative altitude. Relative altitude refers to the altitude of the terrain relative to some reference surface, not relative to sea level. Using relative altitudes avoids problems caused by including altitude values below sea level.

How to deal with the pixels on the edge

In GIS, edge padding refers to filling the boundary of raster data when processing raster data, so that there will be no edge value problems during processing. Usually, when doing raster data processing, such as calculating slope, elevation, etc., the difference between pixels will be involved, and the pixels on the edge have no surrounding pixels to calculate the difference, so it will lead to abnormal edge values or wrong results . In order to avoid this situation, the boundary pixels can be filled to provide the surrounding pixels needed to calculate the boundary pixel value.

When filling boundaries, the usual method is to copy the data. For example, when border-filling raw raster data, you can copy the values of border pixels to outer pixels.

When using GIS software for raster data processing, boundary filling is usually performed automatically to ensure the correctness of the output results. However, if you are doing raster data processing in your own program, you need to do border padding yourself. In

Python, you can use the np.pad() function in NumPy to perform boundary padding. This function inserts each element in an array around the specified number of positions to create a new array.

Use other methods to solve and calculate (GDAL)

GDAL (Geospatial Data Abstraction Library) is an open-source software library that provides a common interface for reading and writing raster and vector geospatial data formats. GDAL provides a powerful set of tools for manipulating geospatial data, including tools for reprojecting and resampling data, extracting subsets of data, merging and splitting data, and more. It can also be used for more advanced analysis, such as calculating terrain slope and aspect, performing raster algebra operations, and generating contour lines

Here is the code to calculate slope aspect using GDAL:

```
# calculate slope and aspect
slope = gdal.DEMProcessing('', dem_dataset, 'slope', computeEdges=True)
aspect = gdal.DEMProcessing('', dem_dataset, 'aspect', computeEdges=True)
```

But in actual operation, it is found that it is not easy to install the GDAL library in Anaconda to use it in Spyder, it also requires the installation of many environments. It is recommended again that you can consider using Colaboratory.

Where it can be improved:

The processing method of loading data can be optimized. What should I do if it is not a txt file but a file in other forms? Is it possible to identify the format and content of the file?

What if multiple data are loaded?

What if the amount of data is large?