Month 1: Watershed Delineation and Terrain analysis

Week 1: Understand DEMs, download SRTM/EU-DEM for area of interest, visualize terrain using rasterio, matplotlib. Read papers on terrain analysis.

---

1. **What's DEM**

   =Raster base Digital Elevation Model (SDTS was the old form of such data)
   =Classified 4 levels of quality (methods of data collection and certainty in the data)
   =Resolution- High resolution smaller cells and more details. Ex: 10m, 30m, 90m etc
   LiDAR or satellite imagery are high resolution
   - 1-arc-second (~30 meters): full coverage
   - 1/3-arc-second (~10 meters): full coverage
   - 1/9-arc-second (~3 meters): partial coverage

   =Horizontal datum of a digital elevation model refers to the reference system used to define the geographic coordinates of the data points in the model. Ensures the data aligns correctly with the Earth's surface.
   =Slope, Aspect and Hillshade

2. **Downloading DEM (for Bavaria):**

   Detail info on various sources: 5 Free Global DEM Data Sources - Digital Elevation Models - GIS Geography

   This is a nice platform where many sources of DEM data are hosted in an organized way. We can download DEM data from here: Home | OpenTopography

- SRTM: This is housed on the USGS earth explorer EarthExplorer
  Bulk download tutorial: How to BULK DOWNLOAD 30m SRTM DEM data from USGS Earth Explorer Website
- Copernicus DEM – Global and European Digital Elevation Model
  Copernicus browser to download DEM: https://browser.dataspace.copernicus.eu/

  Note: Challenge in downloading DEM from these sources that, the DEM come in tiles. In most of the cases our area of interest will not be cover with a single tile, rather a number of tiles. Thus, we need to bulk download and then merge the DEMs to get the whole area covered.
- Python API:
  The elevation tool is a command-line utility that wraps around NASA's SRTM dataset and helps:
  - Download SRTM DEM tiles (automatically selects needed ones)
  - Merge them into a single file
  - Clip to your area of interest
  - Export GeoTIFF files usable in Python, QGIS, ArcGIS

  📦 It downloads 1 arc-second (~30m resolution) DEM data
  Limitation: Again, the tile problem! It cannot download lot of tiles (more than 10) at once to avoid overloading on the server. Thus, we need to divide the area into sections and then merge the tiles using rasterio.
  For now, we don't recommend using python API. Rather we will download DEM manually from web services.

3. **Loading DEMs and merging tiles**
   - Load the raster files (SRTM dem) using rasterio.open() and visualize them individually. We can find more about different raster operations using python following sources:
     - Common raster operations
     - Rasterio: access to geospatial raster data — rasterio 1.5.0.dev documentation

   - Now, if we have multiple rasters covering our area of interest, first, we need to merge them together and write a new raster combining all the tiles. We can use the **merge()** function from rasterio.mask to do that.

     The merge() function returns 2 things:
       1. A NumPy array containing the merger raster data (pixel value)
       2. An affine transformation object that defines the spatial object (Something that we need, and we call it transform)

     After merging, to write the merged raster, we need to retain the transform information and metadata from one of the original rasters (make sure they all have same crs) and use this information to write a new merged raster.

4. **Clipping DEM raster according to watershed**
   - First, load the watershed shape file as a GeoDataFrame and make sure that the shapefile is in the same CRS as our DEM raster.
   - Use the **mask()** function from rasterio.mask to clip the raster. The mask function takes 3 arguments: i) the raster to clip, ii) the geometry of the shapefile iii) crop command (Boolena)

5. **Literature reading:**
   Some basics about terrain analysis

   What does slope, aspect, curvature, hillshade mean?

   - **Slope**: Rate of elevation change; steepness of terrain.
   - **Aspect**: Direction a slope faces, measured in degrees.
   - **Curvature**: Measures concavity/convexity; affects flow acceleration.
   - **Hillshade**: Simulated shading from sunlight angle for visualizing terrain.

   How do they relate to physical processes?

   - **Slope** influences runoff speed and erosion.
   - **Aspect** affects sunlight exposure, soil moisture, and vegetation.
   - **Curvature** controls water convergence/divergence.
   - **Hillshade** helps interpret topographic form visually and in remote sensing.

   Why are they important?

   - Hydrologists use them for water routing; planners use them for hazard assessment; modelers use them for terrain-informed simulations.

Terrain in Hydrologic Processes

How does elevation influence flow direction?

- Water flows from high to low elevation, defining the flow path.

What are flow accumulation and drainage networks?

- **Flow accumulation**: Number of upstream cells draining into a cell; used to derive streams.
- **Drainage network**: Hierarchical pattern of connected flow paths.

How do DEM artifacts affect watershed delineation?

- Artifacts like pits or flat areas mislead flow routing and create false basins or breaks.

🎯 Resolution & Accuracy

What happens when DEMs are too coarse or noisy?

- Terrain features become blurred or misleading, affecting model precision.

How do you correct DEM errors?

- Use **smoothing**, **filtering**, or interpolation to fill voids and reduce noise.

What is sink filling?

- It's a preprocessing step to remove depressions that trap flow in DEMs, ensuring continuous flow routing.

⚙️ Methods and Algorithms

D8 algorithm for flow direction?

- Assigns flow from each cell to the steepest of its 8 neighbors.

Horn's vs Zevenbergen–Thorne method?

- **Horn's** uses weighted neighborhood for smoother slope/aspect; **Z-T** uses a simpler planar fit for curvature-sensitive analysis.

Raster vs vector terrain analysis?

- **Raster** is grid-based and fast for continuous terrain; **vector** uses contours and TINs for detailed, scalable representation.

🌍 Practical Applications

DEMs are used in:

- **Watershed delineation**: to define catchment areas
- **Flood modeling**: to simulate water spread

- **Soil erosion models**: to estimate erosion risk
- **Solar radiation**: to analyze sunlight exposure

Terrain metrics in ML & land use?

- Metrics like slope/aspect/curvature are used as features in landform classification, crop zoning, and terrain-informed models.

<u>Paper 1: Geomorphometry and terrain analysis: data method platforms application</u>

Terrain analysis or Geomorphometry is the quantitative analysis of earth's surface forms using DEMs or other spatial datasets.

Terrain analysis derivatives: Slope, Aspect, Curvature, Flow direction, Topographic wetness index, Hillshade.

Data: Grids/Raster (DEM)- [We are using this in our project], TINs, Point clouds.

Sources: Satellite DEMs: SRTM, ASTER, TanDEM-X

Airborne DEMs: LiDAR, UAV photogrammetry

Derived DEMs: via machine learning, interpolation, downscaling

Challenges: Void area, above ground noise, DEM accuracy

Modern Solution: DEM filtering, Interpolation,

MERIT DEM, FABDEM – Enhanced, hydrologically corrected DEMs.

Analytical Framework: Derivative based – using slope, aspect, curvature.

Feature based – Extracting peaks, valleys, ridged, saddles.

Object based – Segmenting terrain into homogenous units like basins.

[I did not bother to go deeper into the frameworks for now]

Modern methods in terrain analysis:

Machine learning and deep learning: Automated landform classification, DEM enhancement, gap filling etc.

Cloud based terrain analysis: GEE, AWW, Copernicus hub.

Semantic and object-based terrain modelling: multicriteria terrain segmentation, combine topographic and landcover data, classify into meaningful objects rather than raw slope number.

Data Fusion with remote sensing and terrain:

Slope + NDVI – model landslide susceptibility.

Elevation + surface temperature – Detect microclimate zones.

[I need to go deeper into these modern methods]

Week 2: Calculate slope, aspect, hillshade, flow direction/accumulation with whitebox/richdem. Explore flow routing.

For our tasks, both Whitebox and RichDEM can be used. The former one is more used for general geospatial operations while the later one is more specific for hydrology related operations. But RichDEM is a C++ based package which has its limitation of using with python. Thus, we will go with Whitebox.

This channel has many tutorials for whitebox tools: (116) WhiteboxGeospatial - YouTube

And this page describe how to use the whitebox functions: Geomorphometric analysis - WhiteboxTools User Manual

Slope Calculation: One important point is, while calculating slope from a raster, keep in mind that there might be nodata value set as big negative integers in the raster. It can affect the calculation. At this point whitebox can come to help. It can handle the no data value by itself if it is mentioned properly in the clipped raster's meta data.

We do NOT need to fill sink before slope or aspect calculation. We fill sinks before any hydrological routing analysis like flow direction, flow accumulation.

How to analyse slope raster / what to look for in slope raster:

- Slope classification (tentative):

  0°–5°: Flat terrain (likely lowlands, floodplains, urban areas).
  5°–15°: Gentle slopes (agriculture, gentle hills).
  15°–30°: Moderate slopes (transition zones, potential erosion).
  30°+: Steep slopes (hillsides, ridges, possible landslides).

- Where are the steepest areas? Are they clustered around certain features (e.g., rivers, valley sides)?
- Any patterns along edges of the catchment?
- Are your human or land-use features (e.g., settlements) in flatter areas?

Aspect calculation: Shows direction the slope faces, measured in degrees from north

- 0° = North
- 90° = East
- 180° = South
- 270° = West

How to interpret:

- North-facing slopes: cooler, more shaded (in northern hemisphere).
- South-facing slopes: warmer, more sunlight (vegetation/snow melt implications).
- West-facing slopes: afternoon sun, dry faster.

What to look for:

- Where do south-facing slopes dominate? North-facing?
- Are rivers mostly flowing from high-aspect zones to low?
- Could you relate aspect to vegetation, urban layout, or snowmelt?

Hillshade Raster:

A shaded relief image — simulates how light would fall on terrain. Uses your specified azimuth = 270° (west) and altitude = 30° for sunlight direction.

How to interpret:

- Hillshade doesn't carry quantitative data, but it's great for visualizing elevation.
- Highlights valleys, ridges, slopes with shadow and light.
- Used mostly for background visualization.

What to look for:

- Can you visually recognize drainage paths, hilltops, or cliffs?
- Hillshade makes a good base for overlaying vector data (like rivers or catchment boundaries).

Watershed Delineation:

- Fill the sink: There are two whitebox functions for this. wbt.fill_depression and wbt.breach_depression. We used the former one. Difference between these two is available on the wbt documentation page.
- Flow direction: There are many methods to calculate the flow direction raster. We used the most D8 method. wbt has a dedicated function for this method called wbt.d8_pointer.
- Flow accumulation: Again, options are available as different accumulation methods. We used D8 method called wbt.fd8_flow_accumulation.
- Stream extraction: The wbt.extract_streams function is used to extract the streams as a raster. It takes the flow accumulation raster as an input. The threshold has to be specified. It basically depends on desirable resolution. If I want to model a very detailed stream and fine scale, I would set lower threshold.
- Delineation: There are two methods in wbt.
- Basins: wbt.basins() takes the flow direction raster as input and delineate the watershed without any pour/outlet point. One caution is, if the flow direction raster is calculated using wbt, no need to turn on the esri_pntr parameter, otherwise, if it's created by GIS or other tools, turn it on.
- Watershed: wbt.watershed() takes flow direction raster and pour point vector and delineate the watershed.
  While delineating watershed, if our terrain is gentle and DEM is high resolution, the function sees many small drainage-divides and may result in many tiny sub-basins. In this case, we have to use specific points to delineate the watershed.
- Subbasins: subbasins() take flow direction and stream network and divide the watershed into subbasins contributing to each stream.
  When to use what:
  say I have downloaded the DEM for a country, say bangladesh. Now I use basin() on the whole dem to have an idea baout different possible watersheds within the country. And from them I chose one watershed to model rainfall runoff. Then I use watershed() to delineate that specific watershed based on pour points and use subbasins() to delineate based on the existing streams
  Best option: Based on points. So, here comes the step - How to make pour-point vector and use it to delineate.

Vectorization:

A raster can be vectorized by 2 means.

Manually using rasterio.feature.shapes() function to extract shapes from raster, then creating a geodataframe from the shapes.

Using functions from whiteboxtools

Visulization:

Finally visualized the streams, subbasins, gauges on the base of Hillshade.