

NRES 341: Intermediate Geomatics for Natural Resource Management

Paul D. Pickell

2024-12-09

Contents

Welcome	5
How to use these resources	5
How to get involved	5
1 Introduction to Landsat Image Processing	7
Lab Overview	7
Task 1: Data Types & Projections	9
Task 2: Cloud and Shadow Masking	13
Task 3: Image Enhancement and Focal Filters	19
2 Image Classification and Introduction to Google Earth Engine	31
Lab Overview	31
Learning Objectives	32
Deliverables	32
Data	32
Task 1: Getting started with Google Earth Engine	32
Task 2: Calculating NDVI	35
Task 3: Perform a Supervised Classification	36
3 LiDAR Point Clouds for Forest Management	41
Lab Overview	41
Task 1: Load and understand LiDAR data in a Map Project	43
Task 2: Create a DEM, DSM and CHM	45
Task 3: Mapping Tree Tops	48
Task 4: Visualization in 3D	50

4 Terrain and Hydology Analysis	53
Lab Overview	53
Task 1: Understanding DEMs	55
Task 2: Identifying Stream Networks	57
Task 3: Mapping Watersheds	63
5 Mapping suitability and sea level rise	67
Lab Overview	67
Task 1: Mapping flood hazard	69
Task 2: Intersection	74
Task 3: Calculating & visualizing suitability	79

Welcome

These are the course materials for NRES 341 in the Bachelor of Natural Resources program (NRES) at the University of British Columbia (UBC). These Open Educational Resources (OER) were developed to foster the Geomatics Community of Practice that is hosted by the Faculty of Forestry at UBC.

These materials are primarily lab assignments that students enrolled in NRES 341 will complete and submit for credit in the program. Note that much of the data referenced are either public datasets or otherwise only available to students enrolled in the course for credit. Deliverables for these assignments are submitted through the UBC learning management system and only students enrolled in the course may submit these assignments for credit.

How to use these resources

Each “chapter” is a standalone lab assignment designed to be completed over one or two weeks.

Students enrolled in NRES 341 will submit all deliverables through the course management system at UBC for credit and should consult the schedule and deadlines posted there. The casual user can still complete the tutorials step-by-step, but the data that are not already publicly available are not hosted on this website and therefore you will not have access to them.

Unless otherwise noted, all materials are Open Educational Resources (OER) and licensed under a Creative Commons license (CC-BY-SA-4.0). Feel free to share and adapt, just be sure to share with the same license and give credit to the author.

How to get involved

Because this is an open project, we highly encourage contributions from the community. The content is hosted on our GitHub repository and from there

you can open an issue or start a discussion. Feel free to open an issue for any typos, factual discrepancies, bugs, or topics you want to see. We are always looking for great Canadian case studies to share! You can also fork our GitHub repository to explore the source code and take the content offline.

```
## Warning in readLines(file, warn = readLines.warn): incomplete final line found
## on 'chapterauthors.yml'
```

Chapter 1

Introduction to Landsat Image Processing

Written by Hana Travers-Smith

Lab Overview

The Landsat satellite program has been active since 1972 and represents one of the most valuable remote sensing datasets in environmental monitoring and ecology. The Landsat series of satellites measure passive reflectance from the Earth's surface and atmosphere and is used in the fields of agriculture, forestry, geology and hydrology. In 2008, all Landsat data was made open to the public and this has triggered widespread uptake by governments and research groups across the world.

In this lab, you will work with images collected by the Landsat 8 to understand radiometric resolution and get a chance to practice image processing steps including masking clouds and cloud shadows and applying image enhancements.



Learning Objectives

- Understand radiometric resolution and how it relates to Digital Numbers
 - Learn how to resample rasters to common projections
 - Use the Landsat Quality Assurance Band to mask clouds and shadows
 - Understand how image enhancements and focal filters work
-

Deliverables

- Screenshots of image metadata and histograms
 - Answers to 15 questions posed in the handout
-

Data

- Two Landsat 8 Surface Reflectance images
-

Task 1: Data Types & Projections

You are given two Landsat Surface Reflectance images.

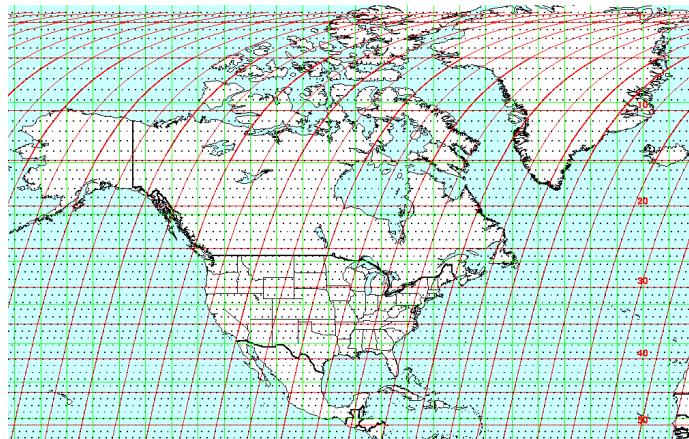
- LC08_047026_20200830_02_T1_a.tif
- LC08_047026_20200830_02_T1_b.tif

The filenames use the following naming pattern that tells you information about the data product and when the image was acquired:
LXSS_PPPRRR_YYYYMMDD_CC_TX.tif

- L = Landsat
- X = OLI/TIRS Sensor
- S = Landsat 8 satellite
- PPP = WRS path
- RRR = WRS row
- YYYYMMDD = Acquisition year, month, day
- CC = Collection number
- TX = Collection Category

WRS path/row refer to a worldwide grid system, where each Landsat scene is assigned a specific path (longitude) and row (latitude) coordinate.

10 CHAPTER 1. INTRODUCTION TO LANDSAT IMAGE PROCESSING

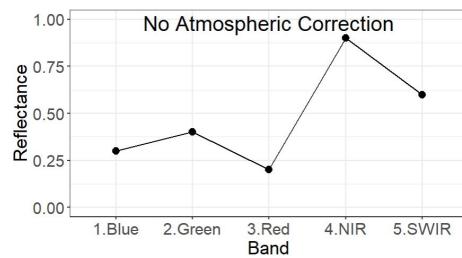


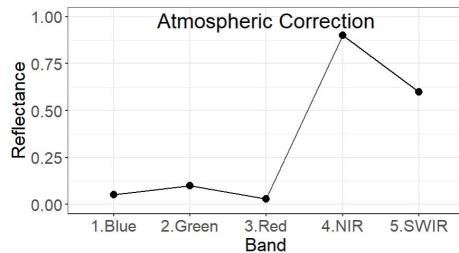
Use the USGS resource to answer the following questions: '<https://www.usgs.gov/faqs/what-naming-convention-landsat-collections-level-1-scenes>'

Q1: For the Landsat scenes you are given, what Landsat sensor and satellite do the images come from? When were the images acquired and what Landsat Collection number are they found in?

Q2: Data in Landsat Collections 1 & 2 have been pre-processed so that images across time are geometrically and radiometrically consistent. In 2-4 sentences explain what this means and why it is important for detecting environmental changes.

Q3: The following image shows the spectral profile of a vegetated surface before and after atmospheric correction. Describe the differences between the two profiles and explain the properties of the atmosphere that causes this. In your own words, why is it important to correct for atmospheric effects when using satellite imagery collected at different times?





Step 1: Import the following rasters **LC08_047026_20200830_SR_A.tif** and **LC08_047026_20200830_SR_B.tif** into a new Map Project in ArcGIS Pro. Name the project Lab 1 and save it in the default documents folder on your computer, typically **C:\Users\YourUsername\Documents\ArcGIS\Projects\Lab2**.

The bands are as follows:

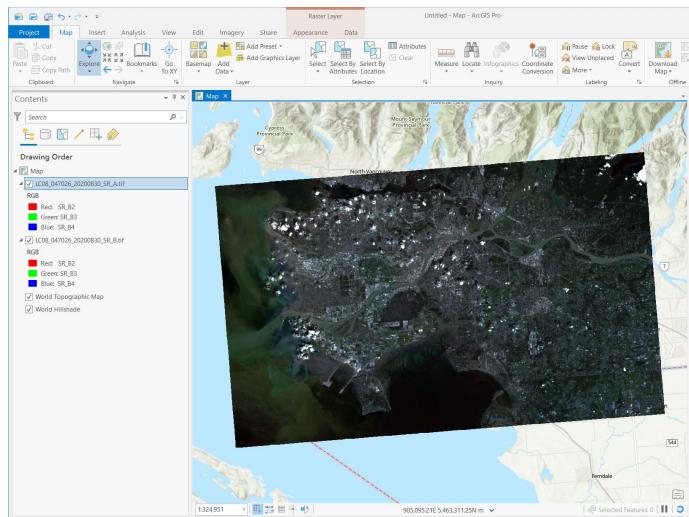
- SR_B2 = Blue
- SR_B3 = Green
- SR_B4 = Red
- SR_B5 = NIR
- SR_B6 = SWIR1

Experiment with the **Symbology** tab.

Screenshot 1: Upload a screenshot of Raster A in RGB true-color.

Screenshot 2: Upload a screenshot of Raster A in false color infrared, with NIR in the red channel. Navigate to > Properties > General and use the image metadata to answer the following questions about the Surface Reflectance rasters.

12 CHAPTER 1. INTRODUCTION TO LANDSAT IMAGE PROCESSING

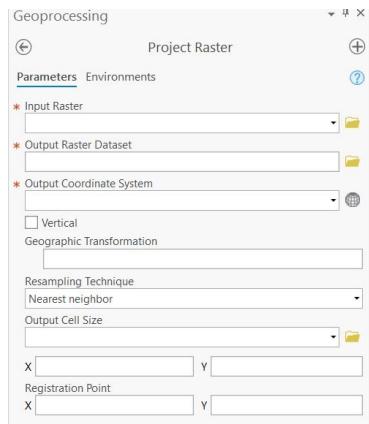


Q4: What are the projections of raster A and raster B? What are the data types?

Q5: Define radiometric resolution and describe how it relates to the range of possible values in the image.

Q6: How many possible values would be present in an 8-bit, 16-bit and 32-bit image? What are the bit-types of rasters A and B (**HINT:** look at the min/max values of the rasters)? Zoom in so you can see individual pixels and notice how the different raster projections change how the pixels align. Raster A is in the correct UTM Zone projection, while raster B is not.

Q7: Imagine you want to see how the reflectance of a small forest stand changes over time. Why would it be important that your imagery is displayed in the same projection? **Step 2:** Convert raster B to the **NAD 1983 UTM Zone 10** projection using the **Project Raster** tool and save the result as a new raster. (Analysis > Tools > search for Project Raster)



Screenshot 3: Upload a screenshot of the Spatial Reference information (in the General tab) for the new raster.

Q8: Which resampling method is most appropriate for continuous data (i.e. temperature, elevation) and why? What about discrete data (i.e. land cover classes, categories)?

Task 2: Cloud and Shadow Masking

Next, we will use the Quality Assurance (QA) band to mask out pixels covered by clouds and cloud shadows.

Landsat (and many other types of remote sensing imagery) use a **Bitmask** to store information related to the quality of a pixel. For each pixel, a bitmask is a series of classifications for whether the pixel contains clouds, snow, shadows, haze, and other atmospheric artefacts we want to remove. Bitmasks also contain information on the level of confidence in the pixel classification. All this information is stored in an integer that can be transformed into its binary counterpart composed of 0's and 1's. Using a bitmask reduces the filesize of a raster, as the integer values are shorter than their binary counterparts.

The Landsat QA_BAND is a 15-bit integer, meaning that the pixel values can range from 0 to 2^{15} . There are 15 different indicators stored in this band that relate to pixel quality (clouds, haze etc...). The full list can be found here (expand the bitmask for QA_PIXEL section): https://developers.google.com/earth-engine/datasets/catalog/LANDSAT_LC08_C02_T1_TOA#bands

14 CHAPTER 1. INTRODUCTION TO LANDSAT IMAGE PROCESSING

QA_PIXEL	30 meters	Landsat Collection 2 QA Bitmask
● Bitmask for QA_PIXEL		
● Bit 0: Fill		
● 0: Image data		
● 1: Fill data		
● Bit 1: Dilated Cloud		
● 0: Cloud is not dilated or no cloud		
● 1: cloud dilation		
● Bit 2: Unused		
● Bit 3: Cloud		
● 0: Cloud confidence is not high		
● 1: High confidence cloud		
● Bit 4: Cloud Shadow		
● 0: Cloud Shadow Confidence is not high		
● 1: High confidence cloud shadow		
● Bit 5: Snow		
● 0: Snow/Ice Confidence is not high		
● 1: High confidence snow cover		

To interpret the pixel value of a bitmask first convert the integer to binary:

For example: 22280 becomes 101011100001000

Starting from the right, each value is assigned a **Bit Position** starting at 0 and counting up to the total number of values.

If we want to know if a pixel is cloudy we need to look at bit 3. In this example, bit 3 is represented by the **fourth number from the right**, and has a value of **1**.

- Bit 3: Cloud
 - 0: Cloud confidence is not high
 - 1: High confidence cloud

According to the bitmask, a value of 1 in bit position 3 indicates a pixel with high confidence cloud.

You will notice that some of the indicators are represented by multiple bit positions. For example, bits 8-9 encode the degree of confidence in the cloud classification.

- Bits 8-9: Cloud Confidence
 - 0: No confidence level set
 - 1: Low confidence
 - 2: Medium confidence
 - 3: High confidence

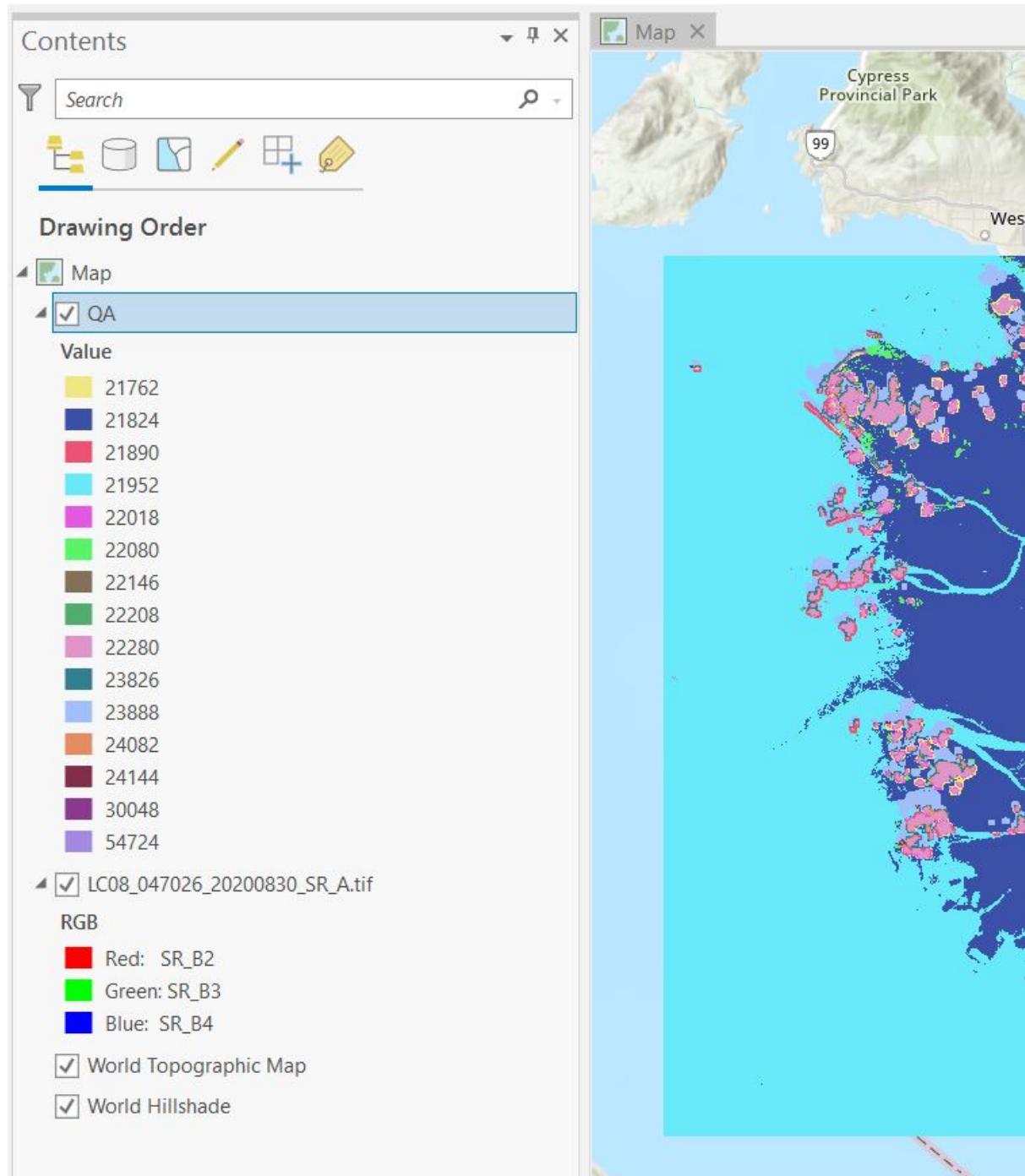
For our example pixel, bit positions 8-9 contain the values 11, converting this binary number back to integer gives the value 3, which corresponds to High confidence.

Use the following online tool to convert between integer and binary numbers and answer the following question: <https://www.rapidtables.com/convert/number/binary-to-decimal.html>

Q9: For a pixel with the integer value 23888, what is the classification for Bit 3: Cloud and Bit 4: Cloud Shadow? What about Bit 10-11: Cloud Shadow Confidence? Step 1: Next, we will create a **mask** representing pixels we want to keep, then use it to remove cloudy pixels from **raster A**. First use the **Make Raster Layer** tool to extract the QA band from the multiband raster.

- Input raster: LC08_047026_20200830_SR_A.tif
- Output raster name: QA_band
- Bands: 6
- leave the other settings as defaults. > Run

Change the **Primary Symbology** of the new QA_band so that each integer is assigned a **unique** color. This will help you see what values represent clear pixels. Your output should look something like the following:



Pixels with clear skies are represented by integers **21824** (land) and **21952**

(water).

Step 2: We will use the **Reclassify (Spatial Analyst)** tool to create a new raster where clear sky pixels have a value of 1 and all other pixels have a value of 0.

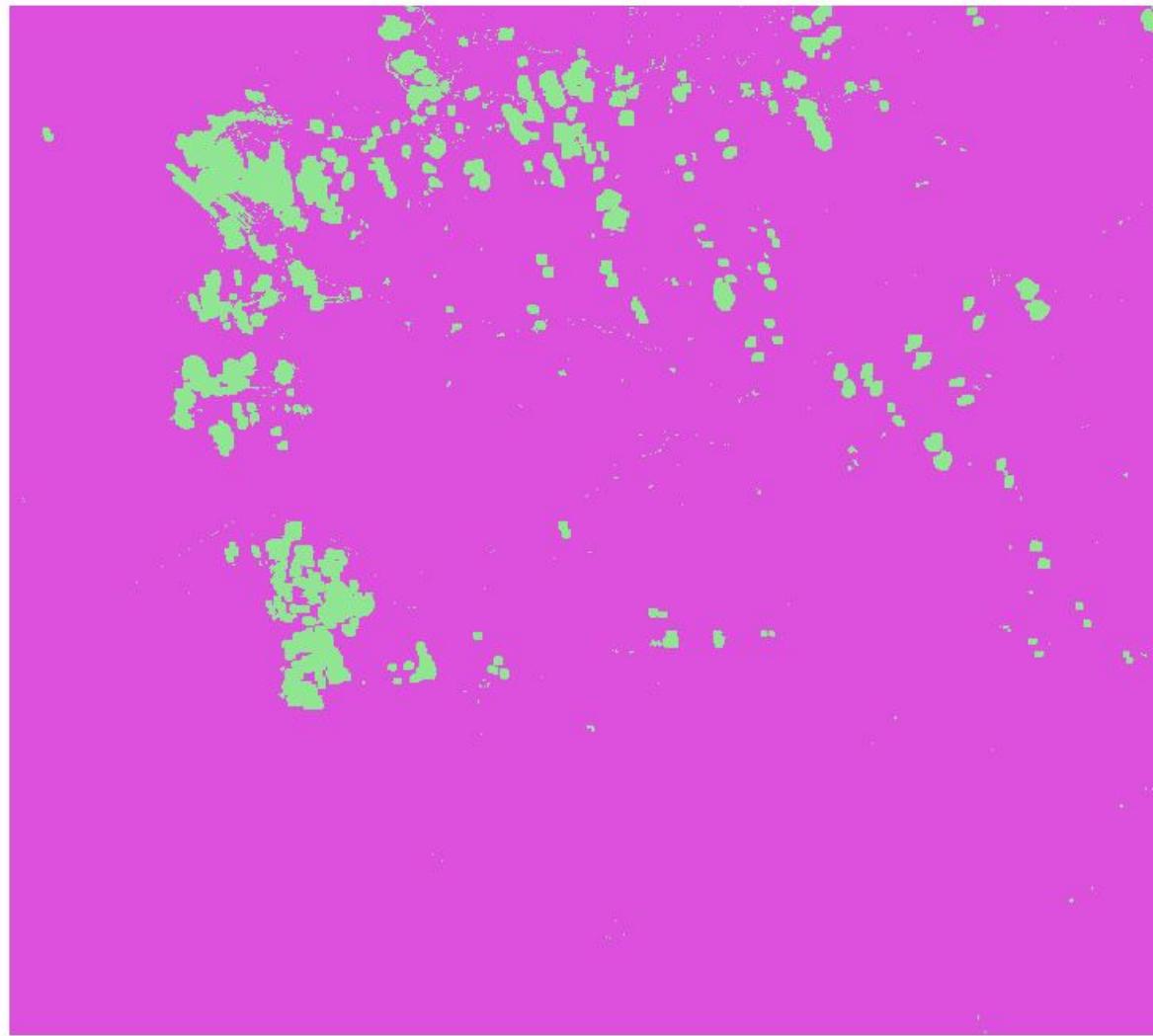
- Input raster: QA_band
- Reclass field: VALUE
- Output raster: Raster_mask

Click **Unique**, for pixel values of 21824 and 21952 set the **New** value to **1** and **0** for everything else.

Reverse New Values	
Value	New
21762	0
21824	1
21890	0
21952	1
22018	0
22080	0
22146	0
22208	0
22280	0
23826	0
23888	0
24082	0
24144	0

You should now have a raster that looks like this:



To create our mask we will convert this raster to polygon features. Navigate to Analysis > Tools > Raster to Polygon.

- Input Raster: Raster_mask
- Field: Value
- Output Polygon Features: Polygon_mask
- UNCHECK Simplify polygons (this will ensure polygons will align with the raster cells)

Click Run.

For the new polygon mask layer, use the **Select by Attributes** tool to select the polygons corresponding to clouds using the expression: `gridcode = 0`, then delete these features in the attribute table. Make sure to save your edits.

Step 3: We are now ready to mask clouds and shadows from the Landsat scene. Navigate to Analysis > Tools > Extract by Mask

- Input raster: **LC08_047026_20200830_SR_A.tif**
- Input raster or feature mask data: **Polygon_mask**
- Output raster: **masked_scene**

Click Run.

Screenshot 4: Upload a screenshot of your masked Landsat scene displayed as false color infrared. Make sure you turn off the basemap layer, so the cloud mask is visible. Note that the bands have been renamed 1-6 corresponding to B, G, R, NIR, SWIR, and the QA band.

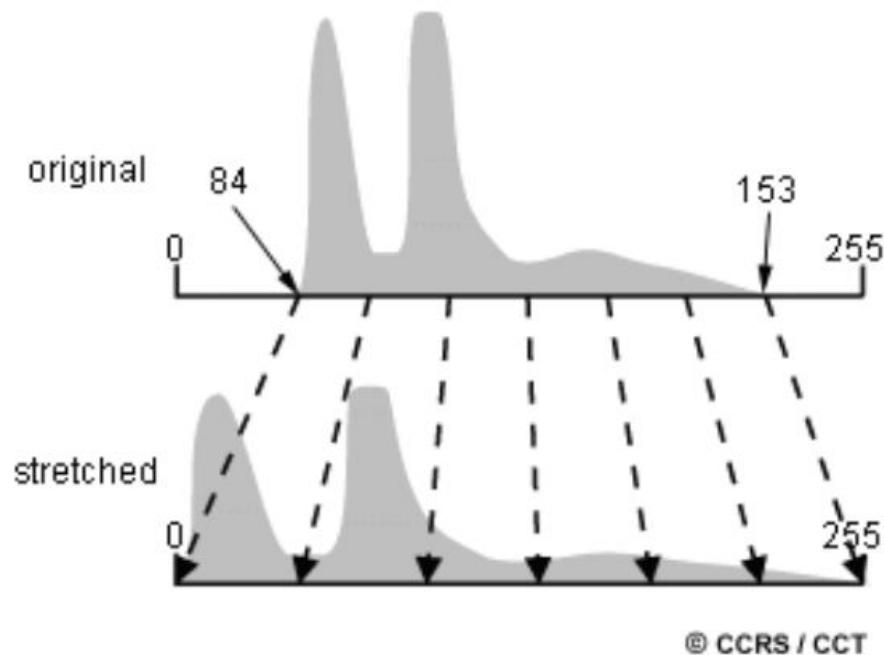
Q10: If you did not have the QA band, what Landsat bands/spectral properties could you use to identify cloudy pixels?

Task 3: Image Enhancement and Focal Filters

Step 1: View the NIR band of the **cloud masked image in greyscale**.

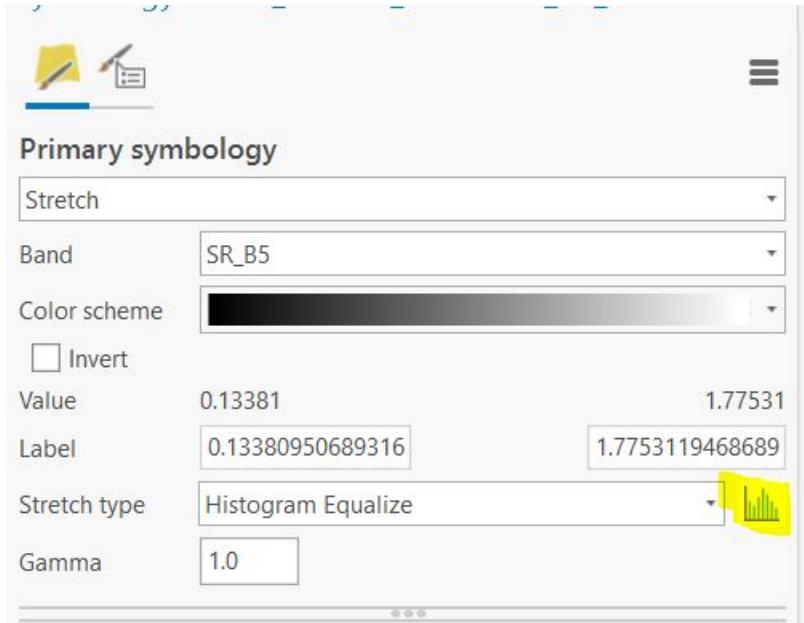
Symbology > change Primary Symbology to Stretch > Set the band to SR_Band4. Notice which land cover types appear bright, and which appear dark.

Q11: What do the light and dark areas of the NIR image represent? How could you use this image to identify vegetation and urban areas? Image enhancement makes it easier to see differences in light/dark areas of an image and aids in visual interpretation. For example, an 8-bit image can contain brightness values that range from 0-255. However, the range of values on a raw image may be smaller (i.e. 50-200), thus this image will have **less contrast** between the darkest and lightest regions, and will appear more homogeneous. A common image enhancement is called a **linear contrast stretch** which remaps the values of an image to cover the full dynamic range. Some image enhancements will first remove the highest and lowest values before stretching to get rid of potential outliers.



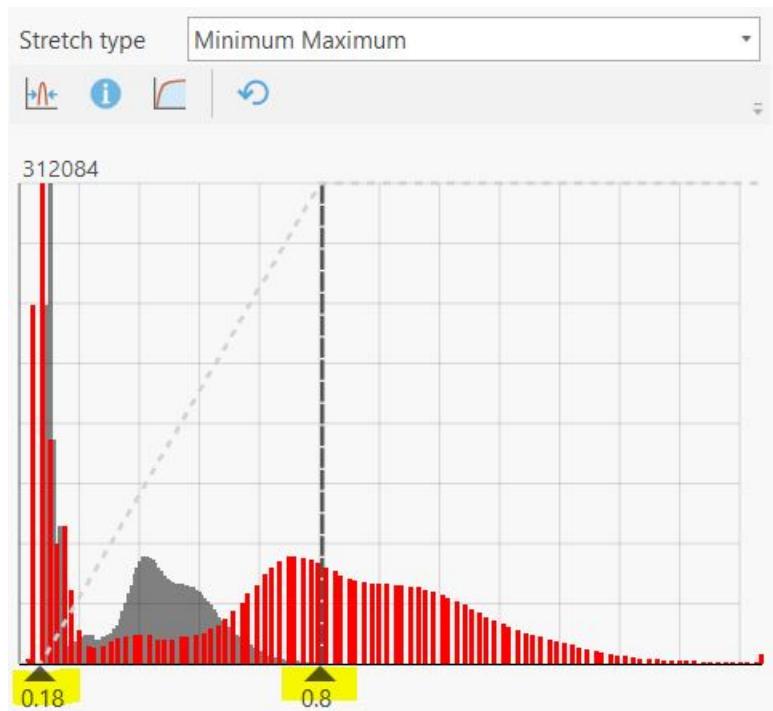
Screenshot 5: Upload a screenshot of the histogram of NIR surface reflectance values. Right-click on the raster layer in the Contents pane > Create Chart > Histogram. Set the variable to the NIR band.

Q12: What is a histogram in the context of a remote sensing image? What do the X and Y axes represent? What is the mean value of the cloud masked NIR image? **Step 2:** Next, we will change the image enhancement in the Symbology tab. Click the histogram symbol.



First, change the stretch type to **Minimum-Maximum** this stretches the image across the full range of values. Next, experiment with moving the sliders along the bottom of the histogram, this will change the minimum and maximum values displayed in the image, and the linear stretch will be applied across this new range. Notice how the image on the map changes.

The grey bars in the background show the original distribution of the values, and the red bars show the stretched values. You can see that applying this transformation increases the range of brightness values across the image.



See the following link for more information about the strech types available in ArcPro: <https://pro.arcgis.com/en/pro-app/latest/help/data/imagery/raster-display-ribbon.htm>

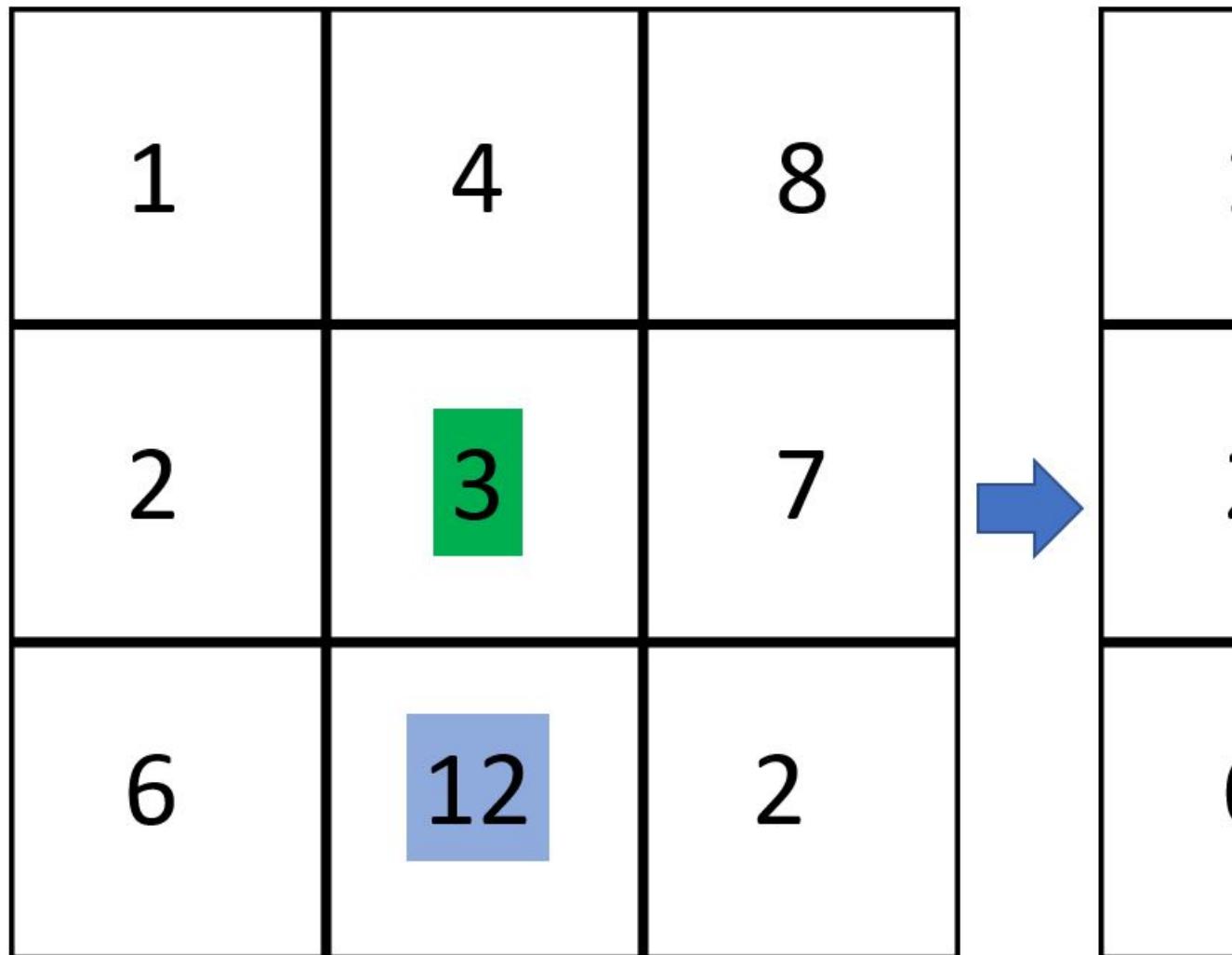
The **Stretch** options are as follows:

- **None**—No stretch method is applied to the layer, even if statistics exist. To display data other than 8-bit data, image values are linearly mapped between 0 and 255. **None** allows you to examine absolute values in your raster datasets.
- **Minimum Maximum**—Applies a linear stretch based on the output minimum and output maximum pixel values, which are used as the endpoints for the histogram. For example, in an 8-bit dataset, the minimum and maximum values can be 33 and 206. A linear stretch is used to distribute the values across 256 values, from 0 to 255. This increases the ability to see differences in values throughout the dataset. **Minimum Maximum** is a good choice for standardizing the appearance of multiple rasters for comparison.
- **Percent Clip**—Cuts off a percentage of the highest and lowest values and applies a linear stretch to the remaining values across the available dynamic range of the data type. This reduces the effects of outliers and enhances the remaining data.
- **Standard Deviation**—Applies a linear stretch between the values defined by the standard deviation (n) value. For example, if you define a standard deviation of 2, the values beyond the second standard deviation become 0 or 255, and the remaining values are linearly stretched between 0 and 255.
- **Histogram Specification**—Applies the specified piecewise histogram to display the pixels values. The piecewise histogram allows you to map the input pixel values to a rendered value.

Q13: Describe how applying a linear stretch changes the appearance of the imagery. What happens as you decrease the maximum value?

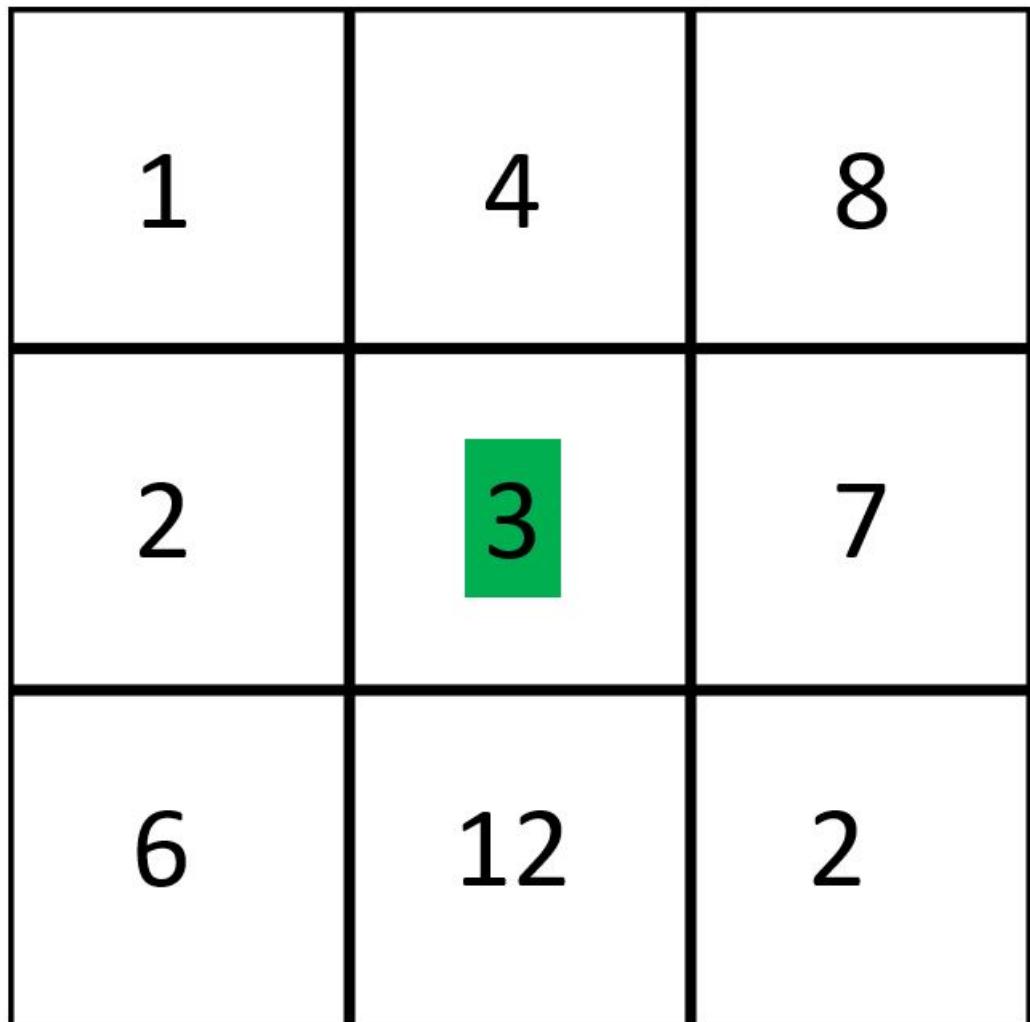
Next, we will explore how **focal filters** work and apply a **Sharpening** and **Smoothing** filter. Focal filters change the pixel values in an image by considering the values of the surrounding pixels.

First, a focal window is defined around a target pixel. In a raster image this is usually the 8 adjacent pixels that touch the target pixel (forming a 3x3 grid). The values of the target pixel and pixels in the focal window are used to calculate a new value for the target pixel. In the example below, the focal filter assigns the target pixel (green) the value of the maximum value in the focal window (blue). The window then moves across the image, and every pixel in the image is transformed to a new value.



Another common filter calculates the **mean** value of the pixels in the window:

Q14: Use the following matrix to calculate the target pixel value (green) using a 3x3 mean filter.



More advanced filters use **weighted** functions, where the pixels in the window are multiplied by a weighting factor that makes them more or less influential in calculating the target value. **Shapening** and **Smoothing** filters use weighted means to enhance or smooth the edges of features in an image. For more information see: <https://desktop.arcgis.com/en/arcmap/latest/manage-data/raster-and-images/convolution-function.htm>

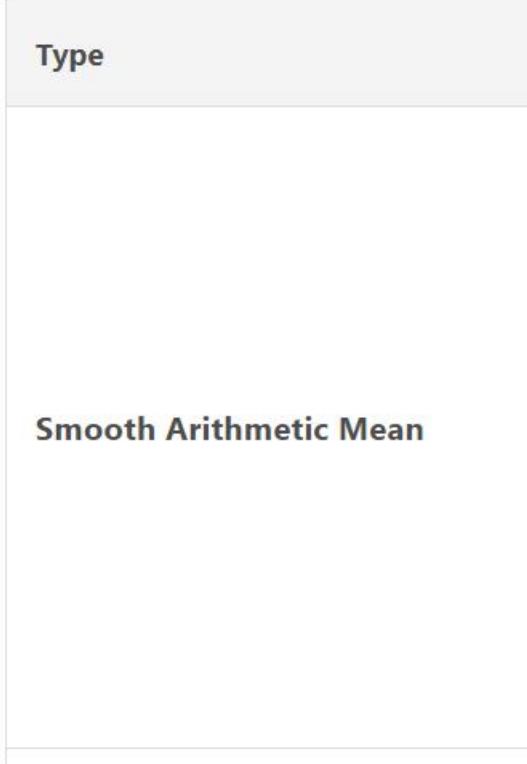
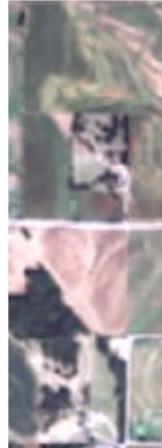
Sharpening types

The Sharpening (high-pass) filter accentuates the comparative difference in the values with calculates the focal sum statistic for each cell of the input using a weighted kernel neighborhood boundaries between features (for example, where a water body meets the forest), thus sharpening the image. The high-pass filter is referred to as an edge enhancement filter. The high-pass filter kernel defines the neighborhood and how much to weight them (multiply them by).

Type	Description	Example
Sharpen	<p>A 3 by 3 filter</p> $\begin{matrix} 0 & -0.25 & 0 \\ -0.25 & 2 & -0.25 \\ 0 & -0.25 & 0 \end{matrix}$	

Smoothing types

Smoothing (low-pass) filters smooth the data by reducing local variation and remeans the average (mean) value for each neighborhood. The effect is that the high and low values are averaged out, reducing the extreme values in the data.

Type	Description	Example
Smooth Arithmetic Mean	A 3 by 3 filter 0.111 0.111 0.111 0.111 0.111 0.111 0.111 0.111 0.111	 

Step 3: Apply a 3x3 **Sharpening** and **Smoothing** filter to the cloud masked image using the **Convolution Tool** in the Raster Functions menu.

Screenshot 6: Upload a screenshot of the sharpened image.

Screenshot 7: Upload a screenshot of the smoothed image.

Q15: Describe how the sharpened and smoothed images differ in terms of the variation in pixel values. Give an example of how each filter might enhance the image interpretation.

```
## Warning in readLines(file, warn = readLines.warn): incomplete final line found
## on 'chapterauthors.yml'
```


Chapter 2

Image Classification and Introduction to Google Earth Engine

Written by Kailey Chelswick

Lab Overview

Wide area remote sensing requires large-scale data storage and computation, which are often limiting factors for studies with large spatial extent or temporal repetition. One solution to the challenge of big data is cloud computing, which allows data storage and processing tasks to happen on remote supercomputing systems (i.e. not on your personal laptop). Google Earth Engine (GEE) provides an interface with which users can efficiently interact with large spatial datasets, including the entire Landsat satellite image archive.

In their 2017 paper, Gorelick et al. describe GEE as “a multi-petabyte analysis-ready data catalog co-located with a high performance, intrinsically parallel computation service”. Essentially, GEE is an online platform that enables analyses on large, internally-stored datasets. This technology has revolutionized the manner in which remote sensing can be conducted, presenting anyone with internet access the opportunity to participate in earth observation science.

Learning Objectives

- Understand applications of land cover mapping
 - Introduction to Google Earth Engine for image processing tasks
 - Learn how to calculate simple spectral indices
 - Use a supervised classification to create a land cover map
-

Deliverables

- Answers to 5 questions posed in the handout
 - Two screenshots of NDVI and the final land cover classification
 - Annotated code used to perform the analysis
-

Data

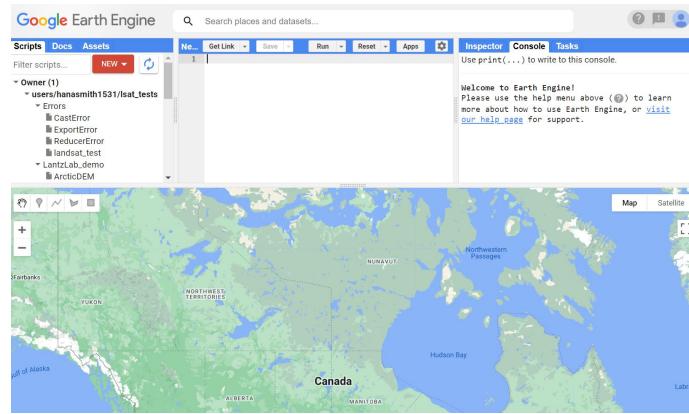
Data will be accessed through the Google Earth Engine code editor. There are thousands of geospatial datasets stored within the GEE catalog. Landsat, Sentinel-1 and many other Earth-observing imagery are included with various levels of pre-processing in order to expedite analyses. Land cover, climate and other relevant environmental datasets are also available.

Task 1: Getting started with Google Earth Engine

Step 1: It is recommended that you use/register for a gmail account prior to applying for access to GEE. To protect your privacy, you can anonymize your account by creating a username that is not attached to your actual name/student ID. Take note of your username and password. Once you have a gmail account, you need to sign-up for GEE using the following link: <https://signup.earthengine.google.com/>

Upon acceptance you will receive a confirmation email. Congratulations, you are now ready to apply your coding and geospatial knowledge in the Earth

Engine Code Editor. GEE runs using Javascript, but don't worry you will be given templates for all the code you will need to complete the lab.

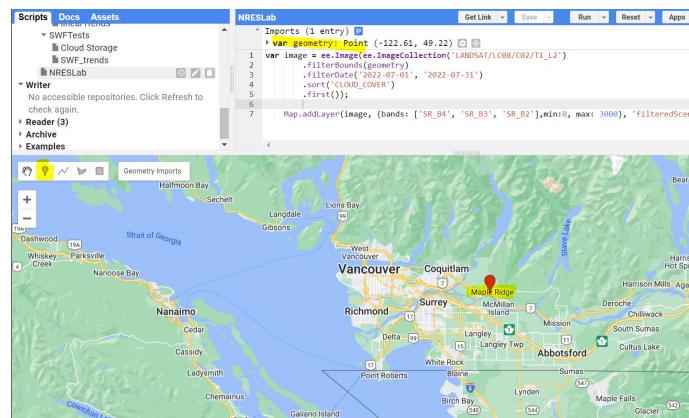


The GEE Code editor is divided into four panels:

- The leftmost panel is where you can access saved scripts, view documentation and access spatial data you have saved to the Cloud.
- The middle panel is where you will write and run your code.
- The right panel is where you will see printed results and outputs.
- The bottom panel shows the map and is where you will display spatial data and imagery.

Step 2: We will practice selecting and displaying a cloud-free Landsat scene over Vancouver. First, we will need to define a Region of Interest to crop our Landsat scene to.

In the Map panel, click on the Pin icon and create a point in Maple Ridge. You will see that this point is automatically named and will appear in the code editor with the variable name ‘geometry’.



The following codeblock selects a cloud-free image from the Landsat 8 Surface Reflectance archive and saves it as a new variable called `image`. Each line in codeblock applies criteria to filter the image archive:

- filter by location - only include images that intersect with the region of interest
- filter by date - include images acquired between two dates
- sort the remaining images by the amount of cloud cover
- finally, select the first image in the list, which will have the least cloudy pixels

The `print` function will print the image metadata in the Console.

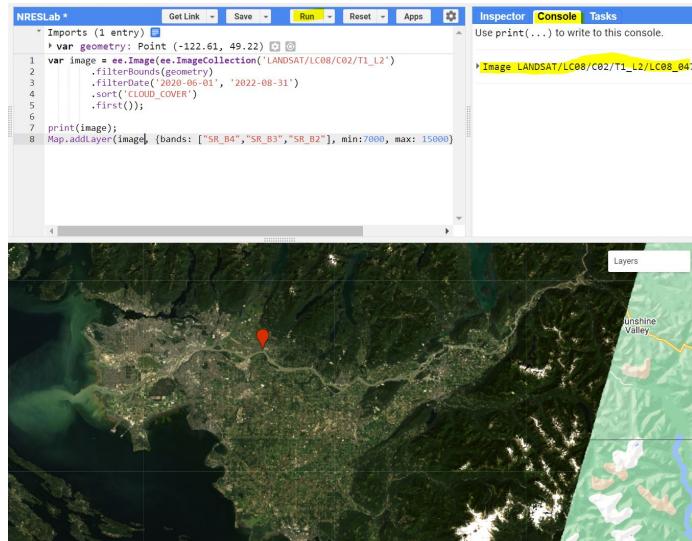
The `Map.addLayer()` function displays the multi-band image in the Map and applies a min/max filter. The `//` symbols allow you to comment the code, these lines do not run anything but can help you organize code and understand what is going on.

```
// Filter an image collection by location, date and cloud cover
var image = ee.Image(ee.ImageCollection('LANDSAT/LC08/C02/T1_L2')
    .filterBounds(geometry)
    .filterDate('2020-06-01', '2022-08-31')
    .sort('CLOUD_COVER')
    .first()); // select the least cloudy image

// Add the image to the Map
print(image);
Map.addLayer(image, {bands: ["SR_B4","SR_B3","SR_B2"], min:7000, max: 15000}, 'filtered')
```

You should now see the following image displayed on the Map.

Expand the image properties in the Console tab and answer the following questions:



Q1: What UTM Zone is the image in? What date was the image acquired? How much cloud cover is present over the land?

Task 2: Calculating NDVI

In this task, you will calculate the Normalized Difference Vegetation Index. In the next task you will use it to perform a supervised classification and produce a map of land cover.

Spectral indices combine spectral reflectance from two or more bands to highlight areas of spectral importance in an image. There are a wide variety of spectral indices used to identify different land covers and image properties including Normalized Difference Vegetation Index (NDVI) Normalized Burn Ratio (NBR) and Normalized Difference Water Index to name a few.

NDVI is calculated with the following formula:

$$NDVI = \frac{(NIR - RED)}{(NIR + RED)}$$

Where NIR is the near-infrared band and red bands. The results of this equation should be between -1 and 1 with values less than 0 representing water and values between 0-1 representing different levels of green vegetation.

Step 1: Use the following codeblock to calculate NDVI using band 5 (NIR) and band 4 (RED), and display it on the map:

```

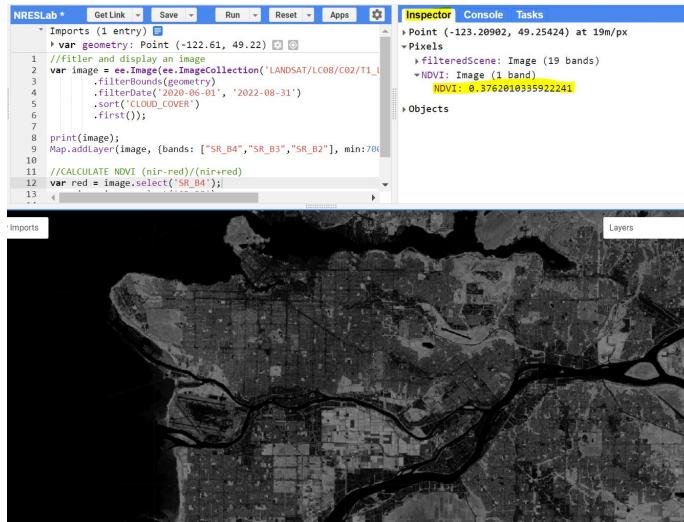
var red = image.select('SR_B4');
var nir = image.select('SR_B5');
var ndvi = nir.subtract(red).divide(nir.add(red)).rename('NDVI');

print(ndvi);
Map.addLayer(ndvi, {min:0, max: 1}, 'NDVI');

```

Screenshot 1: Upload a screenshot of NDVI over the UBC campus.

Q2. Explain how red and NIR reflectance is different for healthy vegetation. How does NDVI use this difference to estimate productivity? Open the Inspector panel and click on pixels in the image to print their values:



Q3. Use the Inspector tool to sample pixels in water, vegetation and urban areas: approximately what range of NDVI values do you observe for these land cover types?

Task 3: Perform a Supervised Classification

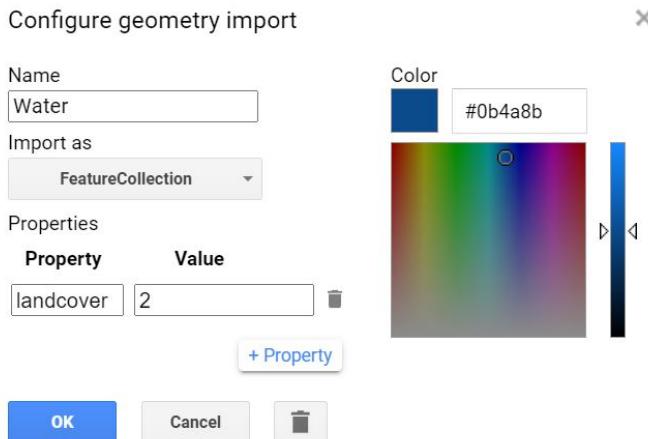
In the section you will use a Supervised Classification to classify the following land cover types:

- Urban

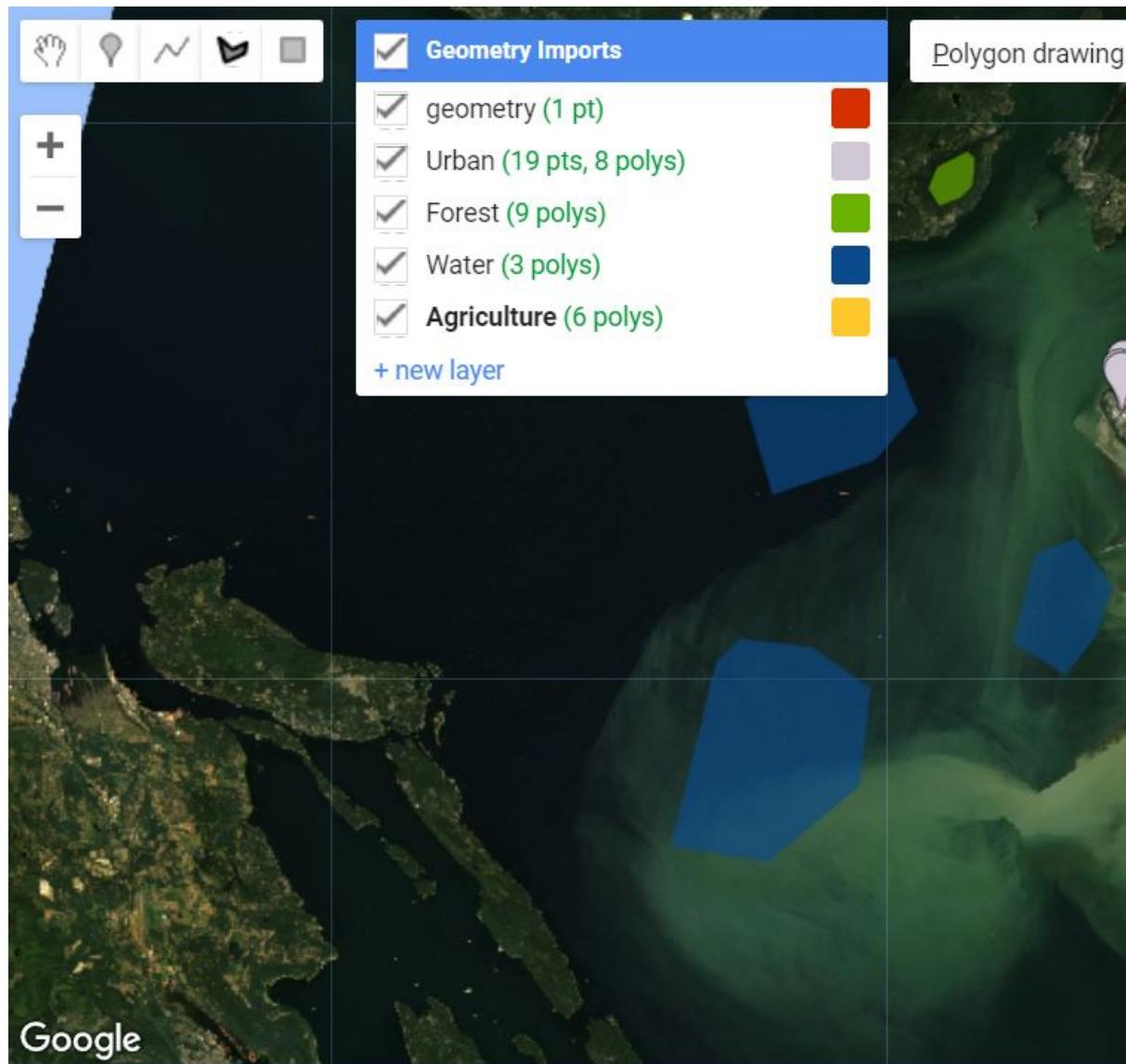
- Water
- Forest
- Agriculture

Step 1: First we will create a sample of pixels to train the classification on. In the Geometry Imports menu, click **+new layer**. Add a new layer for each of the four land cover classes, rename each layer and assign it a color using the gear icon.

In the settings tab (accessed from the gear icon), change Import as to **Feature Collection**. Add a new property called **landcover** and fill it with a **numeric value** unique to each land cover type. For example, Water = 0, Forest = 1, Urban = 2 etc...



Select a land cover class in the Geometry Imports menu and use the point and polygon tools to select pixels in the image representing that class. Toggle the NDVI and true color composite on/off to aid your land cover interpretation. The example below gives an idea of how many pixels you want to sample per class.



Step 2: Once you have selected training areas for each land cover class use the following code block to merge the Feature Collections and select a sample of pixels to train the classification model. The training pixels will be stored as a new variable called `training`.

```
var classNames = Water.merge(Urban).merge(Agriculture).merge(Forest);  
print(classNames);
```

```

var bands = ['SR_B2', 'SR_B3', 'SR_B4', 'SR_B5', 'SR_B6', 'SR_B7', 'NDVI'];
var training = image.select(bands).sampleRegions({
  collection: classNames,
  properties: ['landcover'],
  scale: 30
});

```

Q4: Explain how a supervised and unsupervised classification are different. **Step 3:** Next we will use the **Minimum Distance** algorithm to train our classifier and predict land cover for all the pixels in our image.

First we make a new variable called **classifier** which contains the minimum distance classifier trained on our sample pixels. Conceptually, for a given pixel the model takes into account reflectance from the 6 spectral bands and NDVI and compares it with statistics generated from the training sample. The pixel is then assigned to the land cover class that it is most similar to.

Finally, we apply the model to the entire image and save it as a new layer called **classified**.

```

var classifier = ee.Classifier.minimumDistance().train({
  features: training,
  classProperty: 'landcover',
  inputProperties: bands
});

var classified = image.select(bands).classify(classifier);
Map.addLayer(classified,
{min: 0, max: 3, palette: ['orange', 'green', 'blue','yellow']},
'classification');

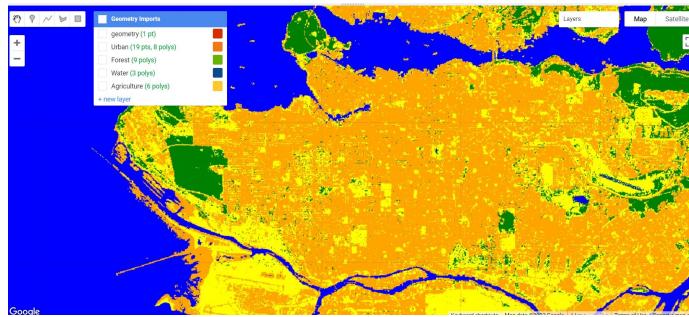
```

Step 4: After you click Run, zoom into the greater Vancouver Area. The new classification layer will take some time to run and be displayed on the map. If you are having trouble viewing the classified layer, zoom in more so that there is less area to compute over.

Depending on the numeric values you assigned each land cover class, you may want to reorder the palette colors so that each color is assigned to an appropriate class

Screenshot 2: Upload a screenshot of the final classified layer (see example below). Include a legend for the land cover types.

Q5: What classes do you think were classified the most and least accurately? Why do you think some classes are more likely to be misclassified?



Annotated Code: Use comments // to annotate each block of code you used to run this analysis. Your annotation should be brief, but explain what each processing step does. Use the first annotated code block for as example.

```
## Warning in readLines(file, warn = readLines.warn): incomplete final line found
## on 'chapterauthors.yml'
```

Chapter 3

LiDAR Point Clouds for Forest Management

Written by Hana Travers-Smith

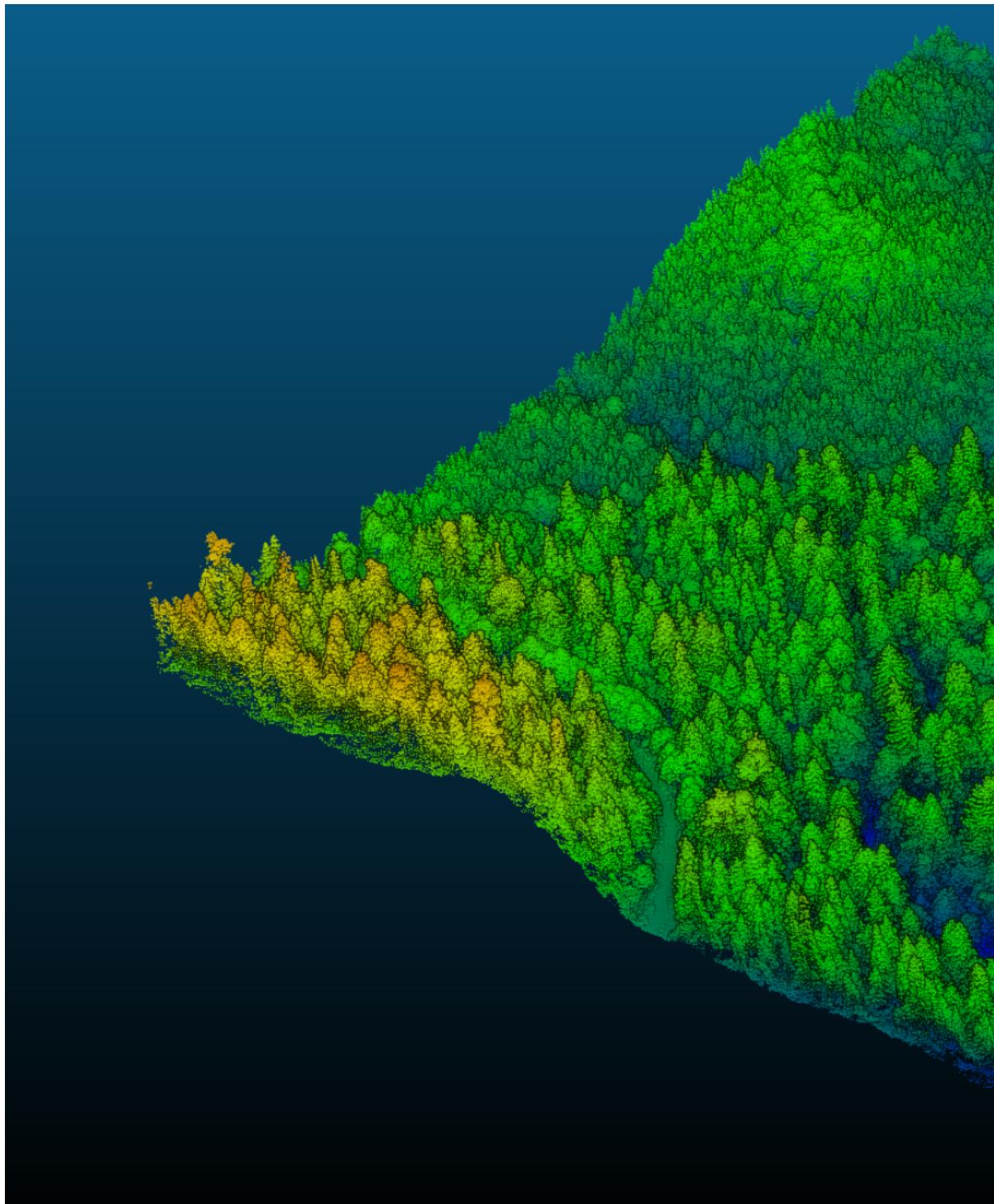
Lab Overview

LiDAR, short for Light Detection and Ranging, is a remote sensing technology that uses laser pulses to measure distances and create detailed 3D maps of objects and environments. To map vegetation, LiDAR emits laser beams from an aircraft or ground-based system towards the vegetation canopy. The laser pulses bounce back upon hitting objects, including leaves, branches, and the ground. By measuring the time it takes for the pulses to return, LiDAR calculates the distance to each point, generating a “point cloud” of data.

“Point clouds” can be used to create high-resolution maps depicting the vertical structure of vegetation, including tree heights, density, and ground cover. These maps are invaluable for understanding ecosystem health, biodiversity, carbon storage, and assisting in land management decisions such as forest monitoring, conservation planning, and urban development.

In this lab you will use high resolution LiDAR data collected by the City of Vancouver to create a model of terrain and vegetation for a forest located on UBC campus.

42 CHAPTER 3. LIDAR POINT CLOUDS FOR FOREST MANAGEMENT



Learning Objectives

- Understand how LiDAR data is collected
 - Learn how to model terrain and vegetation from a LiDAR point cloud
 - Visualize 3D data in a Scene
-

Deliverables

- Screenshots of CHM and final 3D scene
 - Answers to 11 questions posed in the lab
-

Data

Lidar data collected by the City of Vancouver in .las format

Task 1: Load and understand LiDAR data in a Map Project

Step 1: Download the LiDAR data from the following link: <https://opendata.vancouver.ca/explore/dataset/lidar-2018/information/>

The data is split up into a tiled grid system. You will download one tile covering a section of UBC. Use the **Table** view and search for the following tile and download it: **4840E_54550N** by clicking on the .zip link.

Use the metadata from the City of Vancouver data portal to answer the following questions:

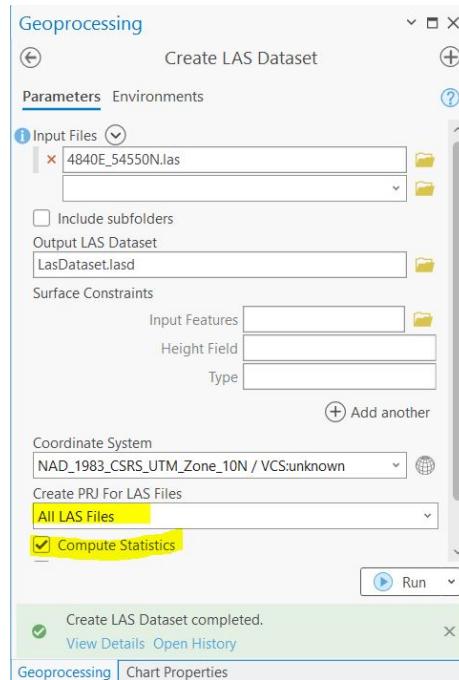
Q1. What is the horizontal and vertical datum of the LAS dataset? Why is it important for Lidar data to have both horizontal AND vertical datums?

Q2: What is the point density of the dataset (points per m²)? Step

2: Create a new ArcGIS Map Project name it Lab5 and save it to the default directory.

ArcGIS Pro has several tools that we can use to view and analyse LiDAR point clouds. In order to view the dataset, we need to import it as a LAS Dataset. Analysis > Tools > type ‘Create LAS dataset’ in the search box.

- Input File: **4840E_54550N.las**
- Output LAS Dataset: **LAS Dataset.lasd**
- Coordinate System: Use the horizontal datum from the dataset specifications
- Create PRJ for LAS Files: **All LAS files**
- Check the ‘Compute Statistics’ box.
- Surface Constraints can be left blank



Step 3: Depending on the zoom extent, you may only see the red bounding box of the las file; this isn’t an error, you just need to zoom in to see the actual points.

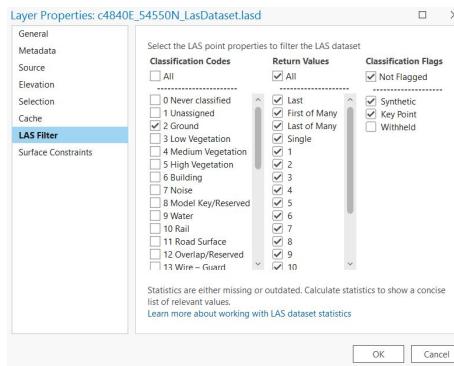
The points can be classified into the following categories:

- Ground

- Non-Ground
- 1st Return (highest feature)

The default display is that no point cloud filters are applied. To quickly filter only ground points right click on the file in the Contents Pane, navigate to ‘LAS Filters’, and click ‘Ground’.

There is also a more detailed classification system, which includes vegetation, water, noise, buildings etc. Right-click on the the file > Properties > LAS Filter. This menu gives you more control for which points you want to display from the dataset.



Task 2: Create a DEM, DSM and CHM

In this task we will create 3 raster datasets by interpolating heights from the point cloud to create continuous surfaces:

- Digital Elevation Model (DEM) - sometimes referred to as a Digital Terrain Model (DTM)
- Digital Surface Model (DSM)
- Canopy Height Model (CHM)

Q3: What does a DEM, DSM and CHM represent? How do you interpret the values in each one? Step 1: First, we will create a DEM from the .lasc point cloud. Filter the point cloud so that only points labelled ‘02 Ground’ are displayed.

Analysis > Tools > Search for LAS Dataset to Raster. We will use the Binning method to interpolate elevation. This method works by dividing the

point cloud into cells (pixels) and assigning a value to each cell based on the heights of all the points in the cell. For example, cell values could be assigned as the average height of all points within a 10x10m pixel.

- Input LAS Dataset: **LAS Dataset.lasd**
- Output Raster: **DEM**
- Value Field: **Elevation**
 - Interpolation Type: **Binning**
 - Cell Assignment: **Average**
 - Void Fill Method: **Linear**
- Output Data Type: **Floating Point**
- Sampling Type: **Cell size**
 - Sampling Value: **10** (this defines the resolution of the output raster as 10m)
 - Z factor: **1**

Q4: What is the minimum and maximum value of the DEM? You should now have something that looks like this:



Step 2: Next, we will create a DSM. We need to ensure that all point labelled as ‘Noise’ are filtered out from the dataset, we will also filter out ‘Ground’ points and just retain points coming from vegetation/infrastructure. Navigate to the LAS Filter menu in the data Properties tab.

Uncheck the following codes:

- 0 Never Classified
- 1 Unassigned

- 2 Ground
- 7 Noise

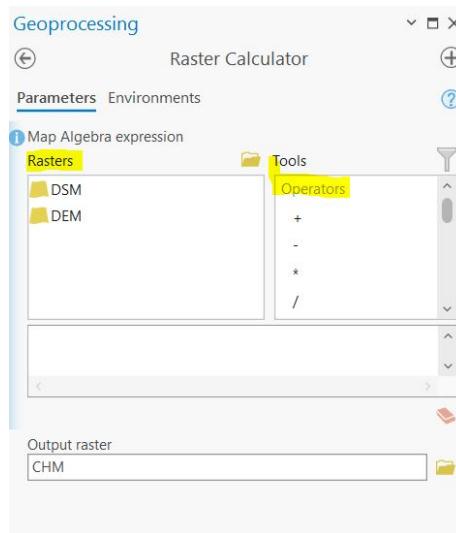
Once you have correctly filtered the point cloud open the **LAS Dataset to Raster** tool. This time we will use the **Maximum** value as the Cell Assignment. Name the output raster **DSM**. All other settings can stay the same as the DEM.
 > Run

Q5: Why might we want to use the Maximum value as the Cell Assignment method for a DSM?

Q6: What features or land cover types can you identify from the DSM?
Step 3: Finally, we will create a CHM using the DEM and DSM. Navigate to Analysis > Tools > Raster Calculator (Image Analyst Tools). This tool allows you to create a new raster by combining multiple rasters using simple mathematical operators (adding, subtracting etc).

Calculate the CHM as: DSM - DEM

Q7: Explain why we need to subtract the DEM from the DSM to calculate canopy height. Describe potential sources of error in deriving a CHM from this method.



Screenshot 1: Upload a screenshot of your final CHM. Change the default symbology to show unvegetated areas, medium vegetation <30m and tall vegetation >30m.

Task 3: Mapping Tree Tops

Using what you learned in the last task, now you will create a point shapefile of treetops using a higher resolution canopy height model.

Step 1: First, we will derive a **DSM** using points representing **High Vegetation**. Open the LAS Filter menu and make sure only points labelled **5 High Vegetation** are checked.

Next, use the **LAS Dataset to Raster** tool to create a raster from the filtered point cloud with the following properties:

- **1m** spatial resolution
- the raster values should represent the highest point in each cell
- name the output **TreeTops**

Q8: What features are now visible in the 1m DSM that were not visible in the 10m DSM? **Step 2:** Next we will use the **Focal Statistics** Tool to identify the maximum height of tree crowns across the forest. We will use a Circular Neighborhood with a Radius of 5. This will calculate the maximum elevation observed within a 5m (5 x 1m) circular moving window. Use the following parameters:

- Input raster: **TreeTops**
- Output raster: **TreeTop_max**
- Neighbourhood: Circle
 - Radius: 5
 - Unit type: Cell
- Statistic Type: Maximum
- Ignore no data in calculations: Checked

Q9: What is a moving window? If you are using a focal maximum, explain how cell values assigned in the final output. **Step 3:** Open the **Raster Calculator** tool. We will use this tool to find the pixels in the **TreeTops** raster that match the maximum focal height in the **TreeTops_max**. To do this, we will use a True/False conditional statement using the Con syntax:

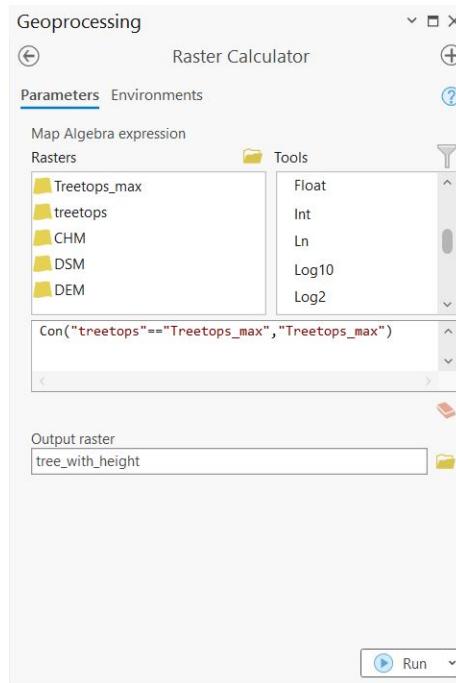
Con(statement, if true, if false) - Essentially for each pixel the statement is evaluated and if it is true an action is taken, and if it is false a different action is taken.

Enter the following statement in the **Raster Calculator** tool:

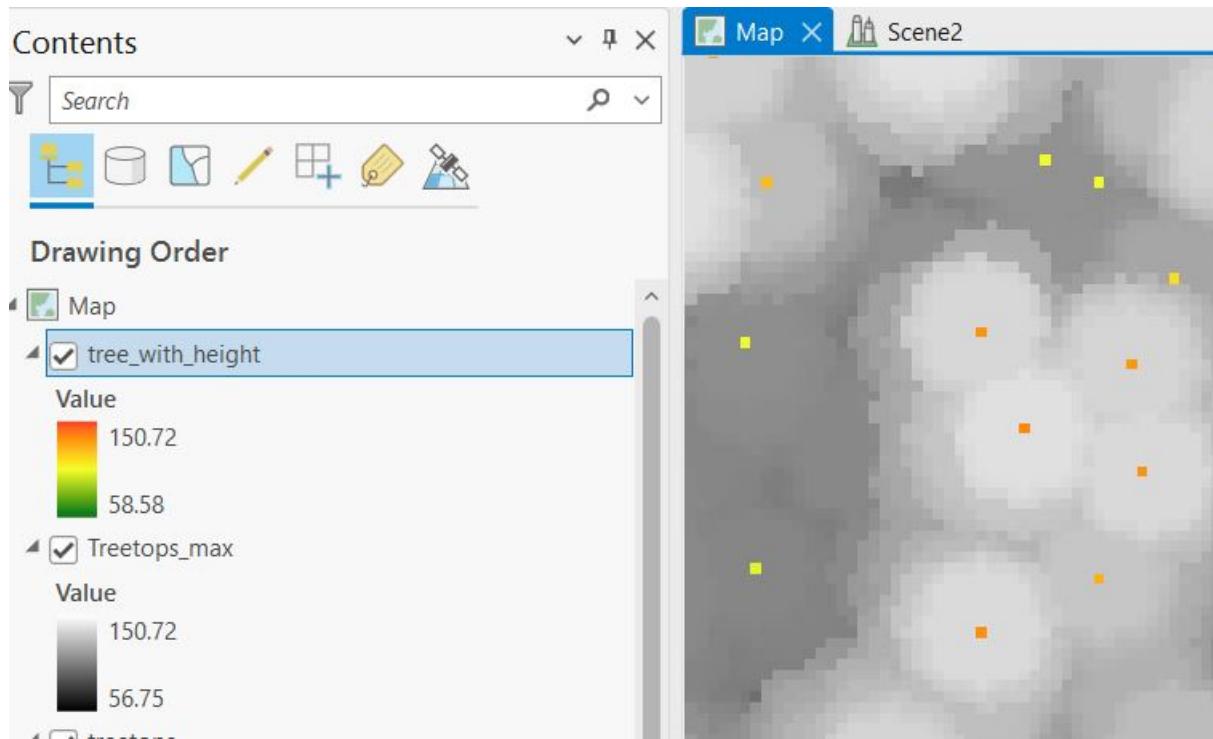
```
Con("TreeTops"=="TreeTops_max", "TreeTops_max")
```

Basically, this is a calculation that evaluates the statement “where is the **Tree-Tops** raster equal to the maximum elevation value identified from the focal statistics?” Where these pixels match, write the maximum value to the output. Otherwise, write a value of NoData to the output. Save the output as **tree_with_height**.

Refer to the screenshot below for writing this statement correctly.



Inspect the output. Zooming in reveals what we have done. Pixels that represent the maximum height in each focal window are assigned a value equal to the maximum height, while all other pixels have a NoData value.



Step 4: Open the ‘Raster to Point’ tool. The input raster is the raster that you just made. Field is Value, and the output name should be ‘tree_with_height’.

- Input: **tree_with_height**
- Field: Value
- Output: **Final_treetops**

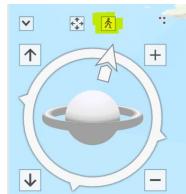
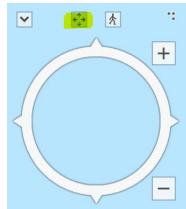
Q10: How many tree tops are in the final output?

Task 4: Visualization in 3D

So far we have worked with the point cloud data in flat, 2-dimensional space. In this task, you will explore the point cloud in 3-dimensions in a Scene.

On the top ribbon Insert > New Map > New Local Scene.

Step 1: Practice using the On Screen Navigator to manipulate the scene. The first toggle let’s you navigate in the normal directions and the second toggle (the person) let’s you rotate the scene in 3D.



Step 2: Navigate to the UBC campus study area and change the Basemap to **Imagery** in the Map tab located on the top ribbon. In the Contents pane, right-click Ground, located below the Elevation Surfaces group layer. Click ‘Add Elevation Source’. Browse to the location of your 10m DSM and select it. You can now start to see how the surface features have been incorporated into the surface. Make sure to zoom in to look at areas with high relief and roads.

Add the **TreeTops_Final** point shapefile to the Scene.

Step 3: Click on ‘Ground’ underneath the Elevation Surfaces tab. On the top ribbon click ‘Elevation Surface Layer’. Set the Vertical Exaggeration to 2 - this will increase the contrast between high/low elevation in the scene. Finally, turn off the WOrldElevation3D/Terrain3d layer to better see how the DSM.

Screenshot 2: Upload a screenshot of the final Scene. Add the tree tops points to the visualization. (See example below, note that yours will also include the tree tops points).

Q11: Experiment with the 10m and 1m DSM. How does the DSM resolution impact the 3D visualization? Which visualization is more realistic?



```
## Warning in readLines(file, warn = readLines.warn): incomplete final line found
## on 'chapterauthors.yml'
```

Chapter 4

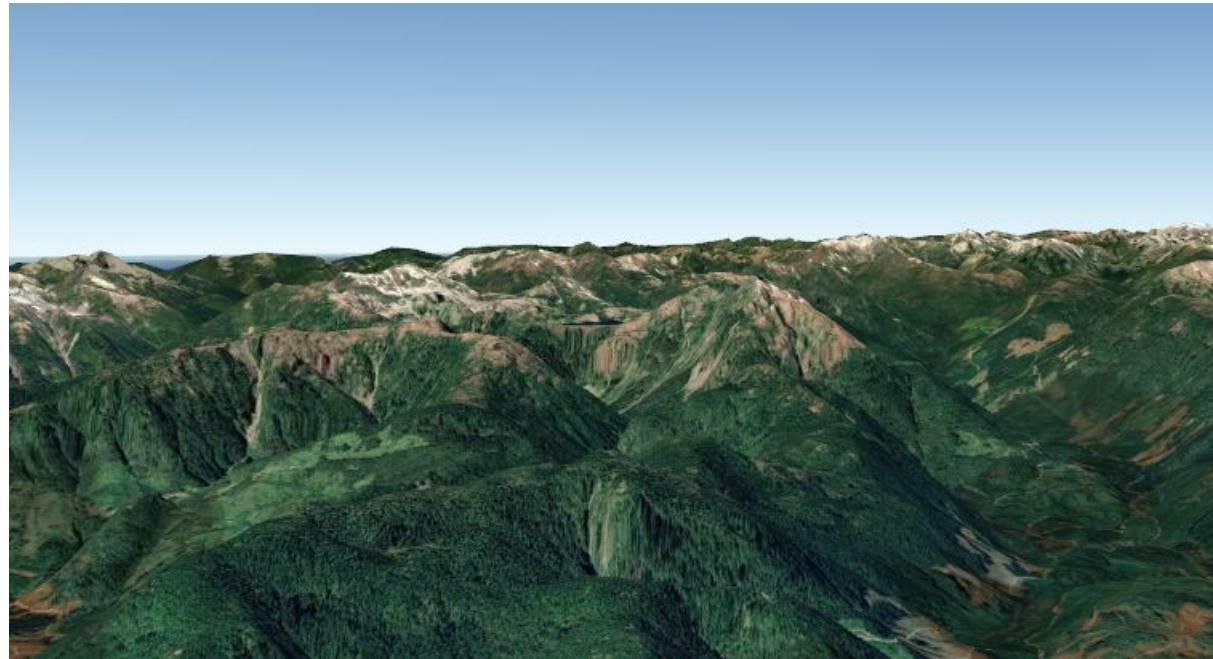
Terrain and Hydology Analysis

Written by Hana Travers-Smith

Lab Overview

A Digital Elevation Model (DEM) is a digital representation of the Earth's terrain including mountains, valleys, rivers, and other topographic features. They are typically created using remote sensing technology, such as radar or LiDAR (Light Detection and Ranging), which capture elevation data points across the landscape. Typically, these elevation data points are organized into raster format, where each raster cell represents elevation within specific pixel. DEMs are used in a range of applications, including cartography, hydrology, geology, environmental analysis, and simulating water flow and erosion.

In this lab you will use a DEM and the Hydrology toolset in ArcGIS Pro to map stream networks and watersheds within critical salmon spawning habitat in Nahmint, BC.



Learning Objectives

- Understand how data is represented in a DEM
 - Learn how to derive slope, aspect and Topographic Position Index (TPI) using raster focal calculations
 - Use the Hydrology Toolbox to map stream networks and watershed boundaries
-

Deliverables

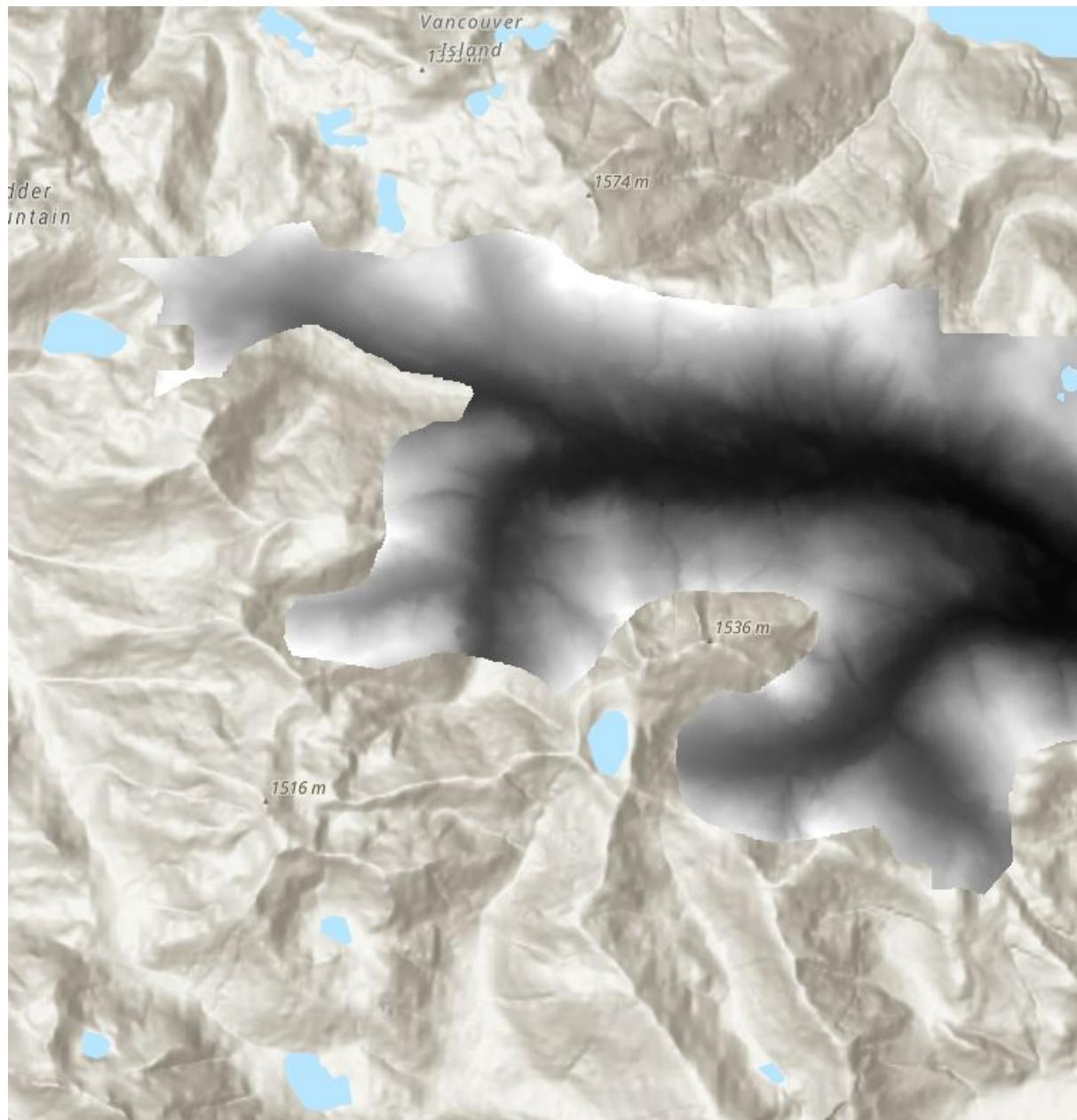
- Answers to 10 questions in the handout
 - A map of the Nahmint watersheds and stream networks
-

Data

- DEM of the Nahmint watershed region, BC
-

Task 1: Understanding DEMs

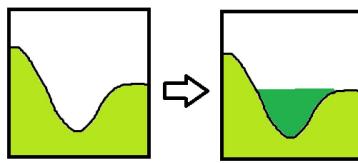
Step 1: Create a new ArcGIS Project name it Lab4 and save it to the default directory. Import the **Nahmint_DTM.tif** and examine the Source information (right-click on the layer in the Catalog pane).



Q1: What is the Projected Coordinate System and spatial resolution of the data?

Q2: What is the Pixel Type and Pixel Depth? How many possible values can be represented by this data? (Report answer as an exponent.)

Q3: What is the difference between a signed and unsigned integer? Which would represent elevation best and why? **Step 2:** First, we will use the **Fill** tool to remove any sinks from the DEM. Sinks are small imperfections in the DEM that create areas where water cannot flow out of. The image below shows the side profile of sink and how its gets filled by the Fill tool. If sinks are not eliminated, water flow can get trapped within these depressions, leading to unrealistic pooling of water and incorrect delineation of watershed boundaries.



Navigate to Analysis > Tools > Fill (Spatial Analyst).

- Input Surface Raster: Nahmint_DTM.tif
- Output Surface Raster: Nahmint_fill
- Z limit: leave blank

Save the output to the default file path (in your ArcMap project).

The Z-limit represents the minimum depth of sinks that will be filled. For example, if it is set to 10m then only sinks deeper than 10m will be filled. For now leave this field blank, this will fill all sinks in the data.

Q4: Why might you want to set a specific z-value?

Task 2: Identifying Stream Networks

Step 1: Next we will use the **Flow Direction** tool to calculate the direction of water flow across the landscape.

There are three flow modelling algorithms, but we will use the simplest: **D8**. In this model water will flow from one cell to its **steepest downslope neighbour**. The cell will then be assigned a value based on which of its 8 neighbours water will flow into.

Q5: The following raster shows elevation above sea level. What is the flow direction from the centre cell? Report your answer in terms of cardinal direction (North, South, Northwest etc)

26	30	16
22	20	14
19	32	11

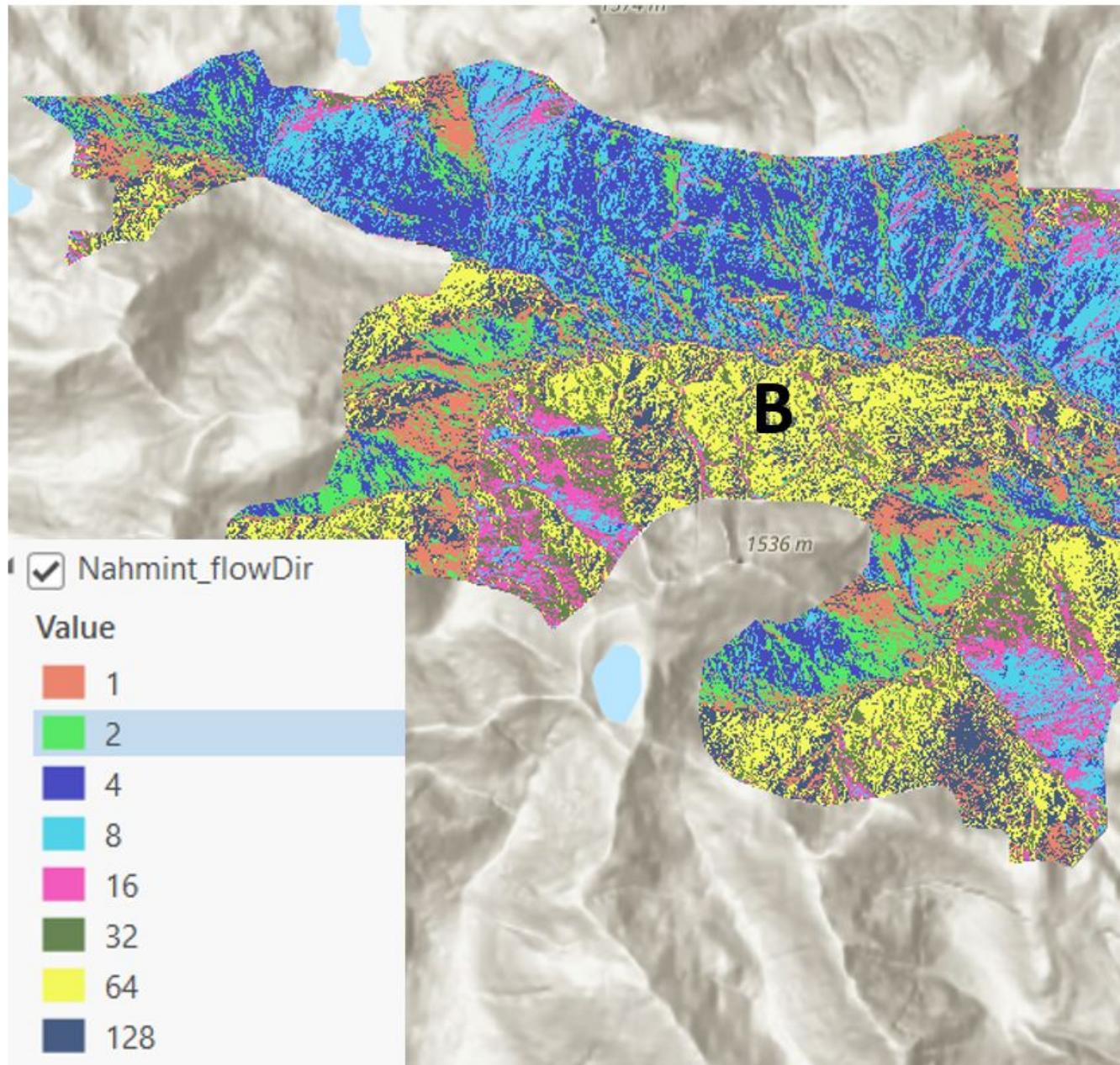
Use the ArcGis help page to answer to following question: <https://pro.arcgis.com/en/pro-app/latest/tool-reference/spatial-analyst/flow-direction.htm>

Q6: If a cell is assigned a Flow Direction value of 32, what cardinal direction is water flowing out of the cell? Navigate to Tools > Search Flow Direction

- Input surface raster: **Nahmint_fill**
- Output flow direction raster: **Nahmint_FlowDir**
- Flow Direction Type: D8
- Leave the rest blank/unchecked

Click Run. You should now have something like the following:

Q7: For the areas marked A and B, approximately what direction is water flowing?



Step 2: We will use the flow direction raster calculated in the last step to calculate **flow accumulation**, which counts the total number of cells that will flow into each cell. For example, a cell located at the bottom of the hill will have high flow accumulation and a cell at the top of a hill will not have any flow.

accumulation.

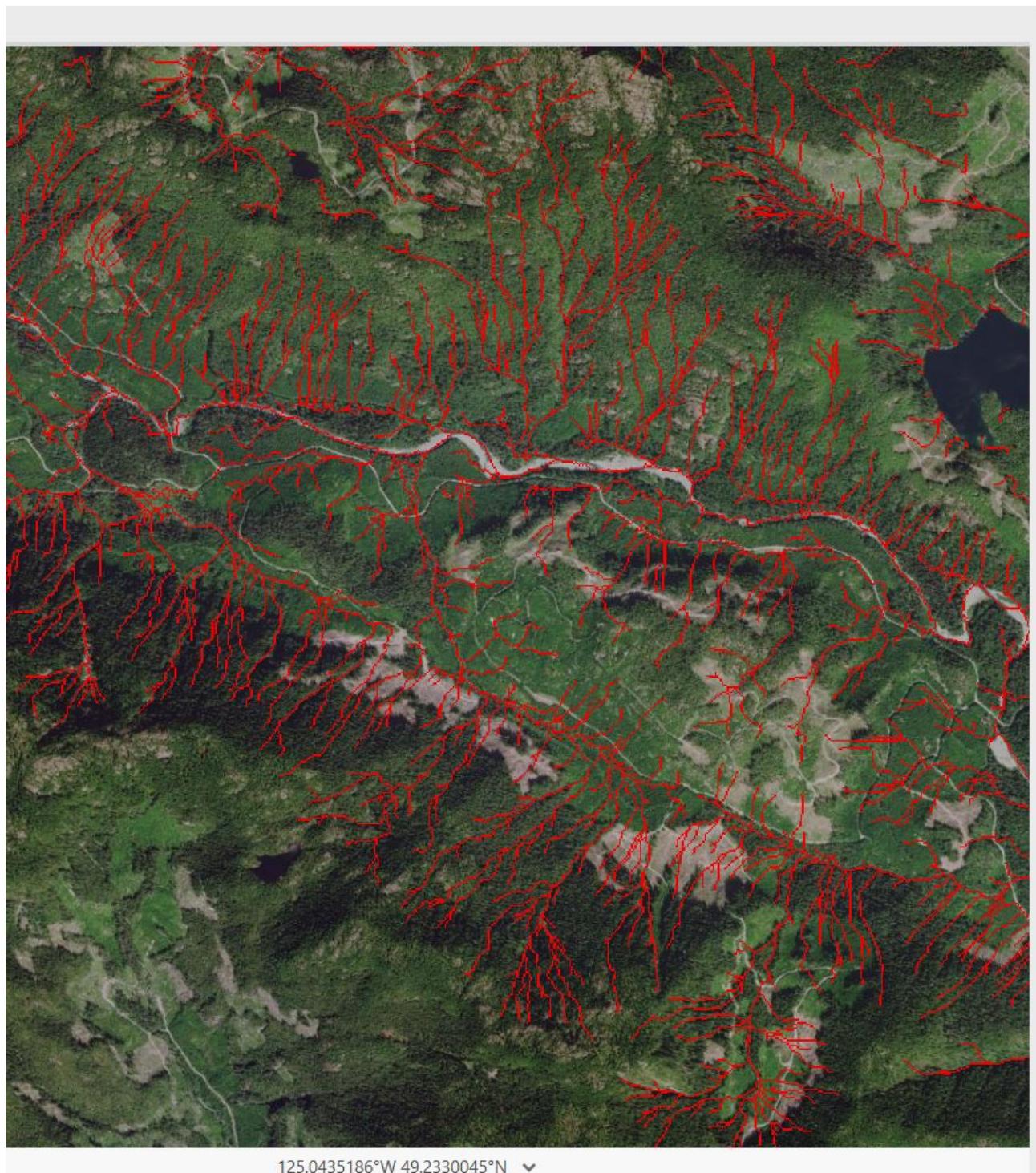
Navigate to Analysis > Tools > Flow Accumulation

- Input flow direction raster: **Nahmint_FlowDir**
- Output flow accumulation raster: **FlowAcc**
- Output data type: **Integer**
- Input flow direction type: **D8**

Leave all other fields blank. > Run.

Step 4: Next, we will create a raster based stream network using a threshold in the flow accumulation raster. For example, if the threshold is 100, then only cells with flow accumulation greater than 100 will be counted as a stream. Cells with flow accumulation less than 100 will be set to a background value of 0.

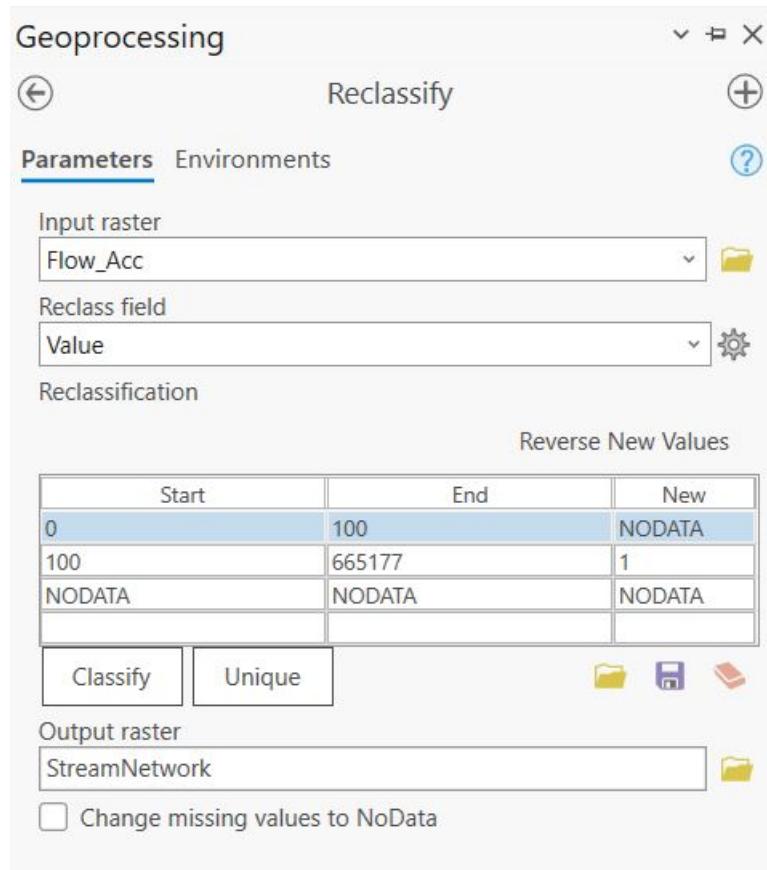
To see how different thresholds impact stream identification, change the **Symbology** of the flow accumulation raster and use the **Manual Interval** symbology to set two classes. See the example below for a stream network with a flow accumulation threshold of 100 (cells with flow accumulation < 100 are set to no color).



Q8: How does the stream network change if you change the threshold from 10, 1000 or 3000? Include a screenshot of each stream network using the different thresholds.

Q9: Compare your stream network to the streams visible in the ArcGIS satellite basemaps. Experiment with different flow accumulation thresholds. Which one seems to represent major streams in the satellite basemaps best? What other land cover/infrastructure in this region may make it difficult to verify smaller stream networks? Once you have selected a threshold, navigate to the **Reclassify** (Spatial Analyst Tools) tool. Use the threshold you have selected as the start and end values. Set the cells representing streams to a new value of 1 and all other cells to NO DATA.

Save the new raster as **StreamNetwork**



Step 5: Finally, we will use the **Stream to Feature** (Spatial Analyst Tools) tool to create polyline features representing our stream network. This

tool uses the stream network and the flow direction layers. See this ArcGIS help page for more information: <https://pro.arcgis.com/en/pro-app/latest/tool-reference/spatial-analyst/stream-to-feature.htm>

Navigate to Analysis > Tools > Stream to Feature

- Input stream raster: **StreamNetwork**
 - Input flow direction raster: **Nahmint_FlowDir**
 - Output polyline features: **StreamNetwork_lines**
 - Simplify polylines: Checked
-

Task 3: Mapping Watersheds

A watershed is an area of land where all the water that falls or flows into it converges to a common outlet, such as a river, lake, or ocean. It is bounded by a topographic or drainage divide, which separates water flowing into different basins. In this task we will delineate the boundaries of the Nahmint watershed.

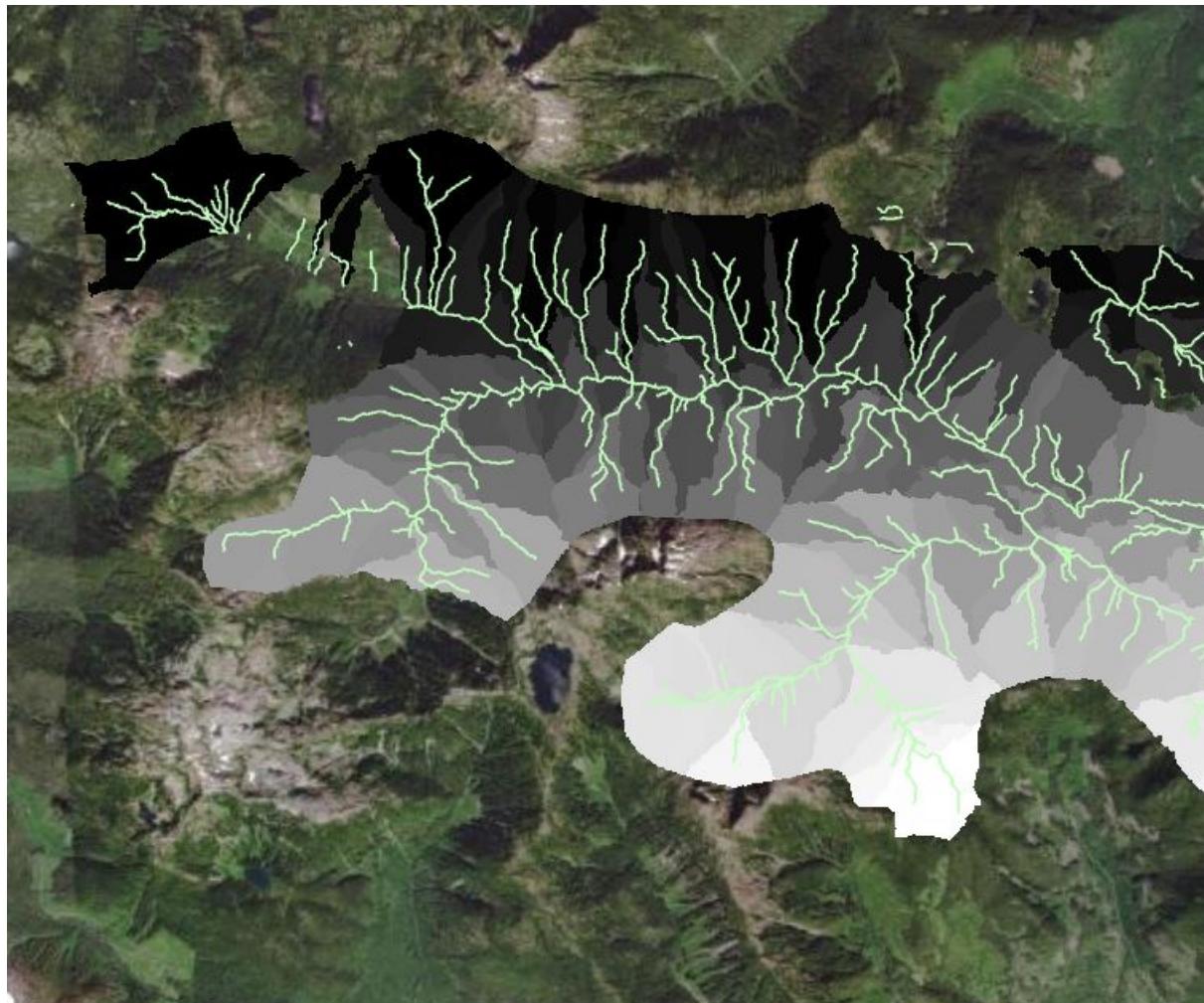
The watershed tool uses flow direction and **stream links** to delineate watershed boundaries. Stream links represent the individual stream segments that make up the entire network. The watershed boundaries will be defined such that water flows into each of the stream links.

Step 1: First, we will use the **Stream Link** tool to generate the links. Use the **raster** stream network and the flow direction raster as the inputs. Save the new raster as **StreamLinks** > Run.

Step 2: Navigate to Analysis > Tools > Watershed

- Input D8 flow direction raster: **Nahmint_FlowDir**
- Input raster or feature pour point data: **StreamLinks**
- Output raster: **Nahmint_watersheds**

The output will be a new raster where the cell values correspond to each unique watershed catchment.



Q10: How many unique watersheds did you define? What is the area of the largest watershed area? Report your answer in km² and round to 2 decimal places. **HINT:** Examine the raster attribute table to estimate watershed area. **Step 4:** Create a map and include it in the final deliverables. The map must have the following elements:

- Stream network polylines
- Watershed polygons - assign different colors to each polygon
- Title
- North arrow
- Scale bar
- Legend

```
## Warning in readLines(file, warn = readLines.warn): incomplete final line found
## on 'chaptersauthors.yml'
```


Chapter 5

Mapping suitability and sea level rise

Written by Hana Travers-Smith

Lab Overview

Climate change is rapidly altering the stability of the Earth's cryosphere. Over the next century thawing glaciers and ice sheets are expected to result in sea level rise between 2-5 m. This is a concern for many coastal environments and communities.

In this lab you will use a digital elevation model (DEM) covering the metro Vancouver region to map areas at risk of flooding during an extreme weather event under current and future conditions, assuming 2.96 m of sea level rise by the year 2100. In particular, we will focus on mapping the intersection of high risk areas with critical agricultural land in the Fraser Delta region. In this assignment, you will produce a map to assess regions where agricultural land can be developed safely from flood hazards.

In partnership with the Delta community, the CALP research group at UBC produced the following report on projected sea level rise and adaptation strategies for this region, which can be found here:

https://www.fraserbasin.bc.ca/_Library/CCAQ_BCRAC/bcrac_delta_visioning-policy_4d.pdf



Learning Objectives

- Understand how to use basic spatial analysis tools (Clip, Intersect, Buffer)
 - Experiment with visualization of spatial layers
 - Conduct a suitability analysis to determine where to expand new agricultural land
-

Deliverables

- Answers to 13 questions
 - Map showing suitability of agricultural areas for future development
-

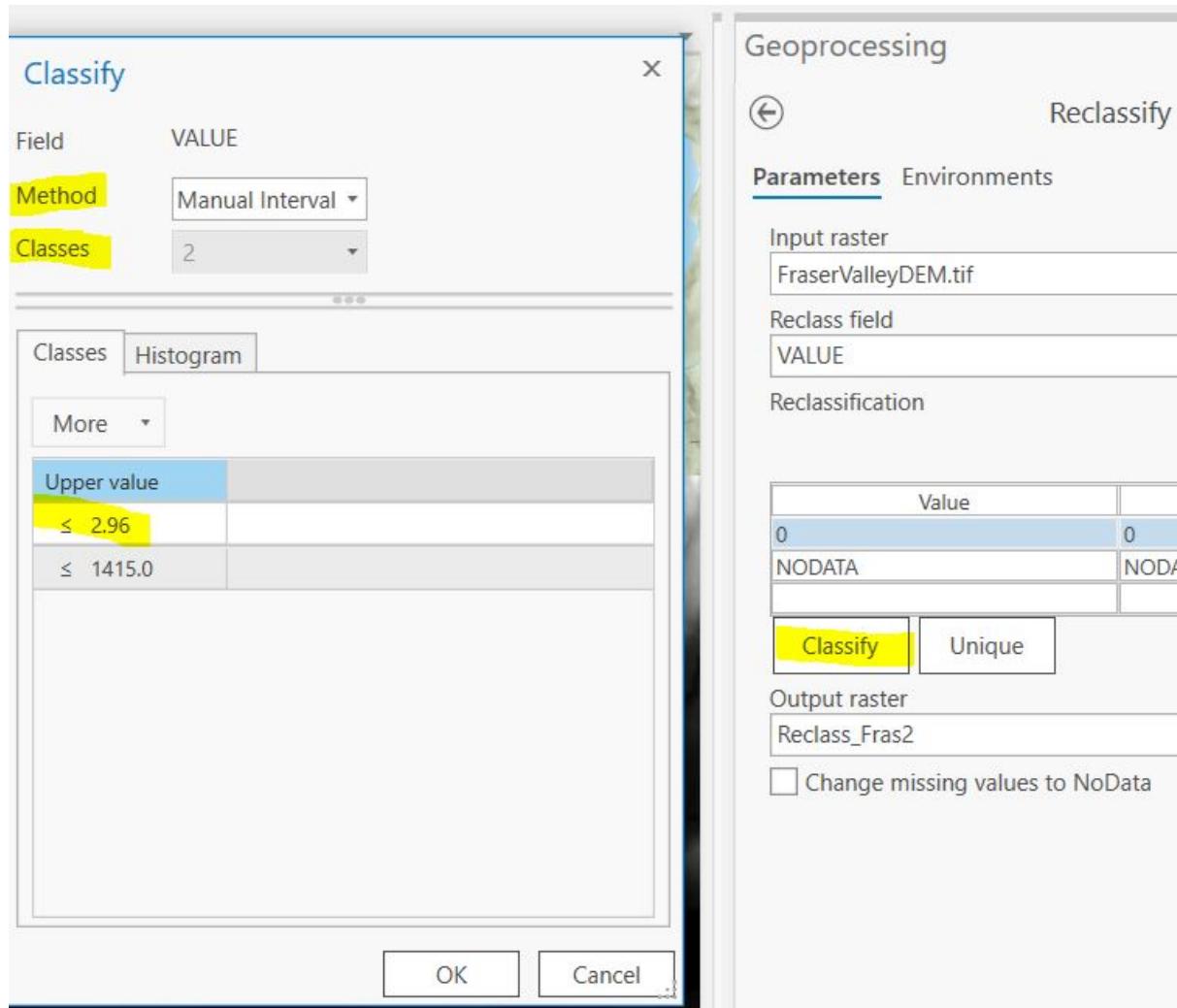
Data

- Digital Elevation Model of the Fraser Valley: **FraserValleyDEM.tif**
 - Polygon of study area: **fraserValley_studyarea.shp**
 - Polyline of coastline: **coastline.shp**
 - Polygon shapefile of agricultural land from <https://catalogue.data.gov.bc.ca/dataset/alc-alr-polygons>: **ag_polys.shp**
-

Task 1: Mapping flood hazard

Peak water levels during a storm are expected to reach **2.96m** above current sea level by 2100 (CALP, 2012). In this lab, we will assess flooding in the delta region to inform future land use planning. For now, we will map flood hazard assuming sea walls and dyke infrastructure is not in place, and there are no obstructions blocking the flow of water over the land.

Q1: From the CALP report (Section 2), briefly list the 4 primary drivers of sea level rise? Step 1: Set up a new Map Project and import the DEM into ArcGIS Pro. First we will identify cells in the DEM that are below the projected high water line of 2.96m. Open the **FraserValleyDEM.tif** in ArcGIS Pro. Navigate to **Analysis > Tools > Reclassify (Spatial Analyst tools)**.



Click the Classify button and in the pop-up window set the number of classes to 2 and the method to Manual Interval. Set the Upper value to 2.96 and hit OK. In the Output raster field give the new raster a descriptive name (ie DEM_reclass_2m) and save it to the default GDB.

HINT: Giving concise and descriptive names to your spatial data will make things easier to keep track of later on! Note that names should not include periods or spaces

RECLASSIFY RASTER VALUES		
Start	End	New
0	2.96	1
2.96	1415	2
NODATA	NODATA	NODATA
Classify	Unique	

Cells in the resulting raster with a value <2.96 will be assigned a new value of 1 and cells >2.96 will be assigned a value of 2.

Q2. What is the spatial resolution of the DEM? Step 2: Next, we will convert the reclassified raster into polygon features. Navigate to the **Raster to Polygon Tool (Conversion Tools)**. Set the input raster to the reclassified raster (DEM_reclass_2m) from the previous step and name the features **flood_2m_polys** and save to the default GDB. Make sure the **Simplify Polygons** box is checked.

Q3: What would happen if we did not simplify the polygon features? Why might we want to produce simplified polygon features? Step 3: Open the attribute table for the polygon features. The **Gridcode** variable corresponds to the cell values from the input raster. In this case we are only interested in keeping polygons representing low elevation cells. We will delete high elevation polygons by selecting features with gridcode = 2

Click **Select by Attributes** and use the drop-down menus to generate the following expression:

Where, gridcode, is equal to, 2

Click the **Delete Selection** button to delete the selected polygons.

InundatedArea_2m

Field: Add Calculate Selection: Select By Attributes Zoom To Switch Clear D						
	OBJECTID *	Shape *	Id	gridcode	Shape_Length	Shape_Area
1	1	Polygon	1	1	112.494012	576.342773
2	2	Polygon	2	1	120	900
3	3	Polygon	3	1	269.983098	4089.465165
4	4	Polygon	4	1	120	900
5	5	Polygon	5	1	180	1800
6	6	Polygon	6	1	240	2700
7	7	Polygon	7	1	120	900
8	8	Polygon	8	1	112.494865	576.34964
9	9	Polygon	9	1	120	900
10	10	Polygon	10	1	113.187071	616.207681
11	11	Polygon	11	1	120	900
12	12	Polygon	12	1	120	900

|◀ ▶| 12 of 3,512 selected

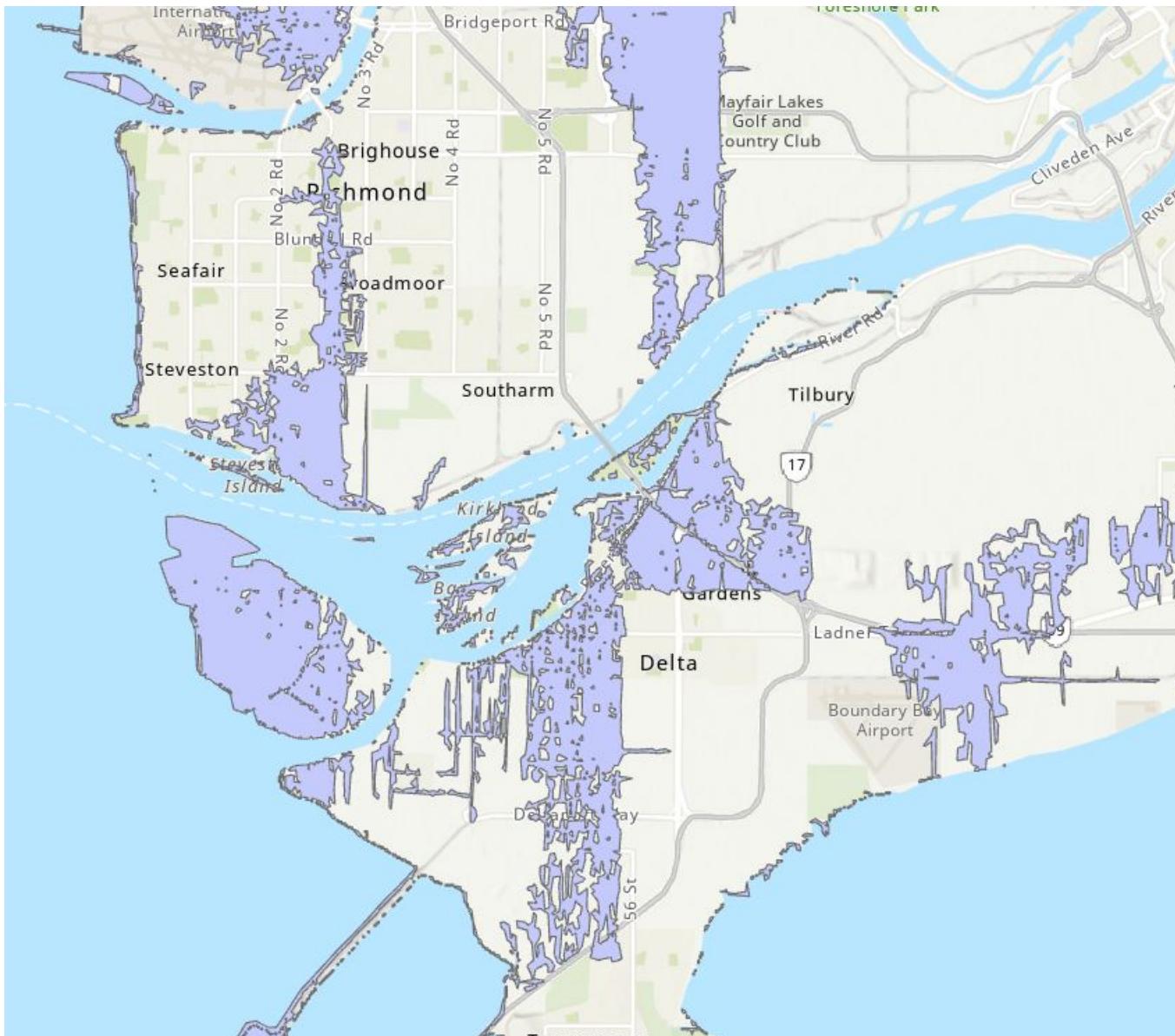
Step 4: Examine the resulting polygon shapefile. You will notice that there are lots of small isolated polygons, not adjacent to the coastline. We will remove these by intersecting the flooded areas features with a polyline representing the coastline.

Open **Coastline.shp**. Navigate to **Map > Select by Location**. Set the Input Features to the **flooded areas polygons** and the Selecting Features to the **coastline layer**. Set the Relationship to **Within a distance**. Set the distance to 30m. **Make sure the Invert spatial relationship box is checked**. This will select polygons that are further than 30m from the coast. Click **Apply**.

Delete the selected features.

Q4: Describe how you would change the selection parameters (Relationship & Distance) if you wanted to select flooded areas that are

within 100m from the coastline? You should now have polygon features that look like the following:



Step 5: Zoom into the polygons. You will notice that even though we used simplified polygons, the edges are still quite jagged and have many small holes. Next, we will smooth these polygons for better visualization and create more

realistic shapes.

First, we will reduce the jagged edges in the polygons. Navigate to the **Buffer** tool. Create a 30m buffer around the **flood_2m_polys** layer. Set the Dissolve type to **Dissolve all output features to a single feature**. Name the output **flood_2m_30mBuffer** Click Run.

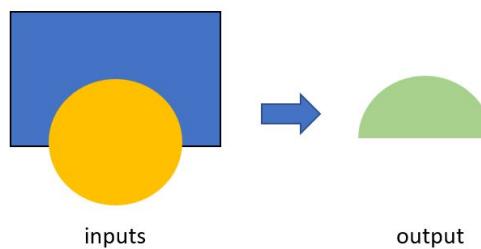
Q5: Describe what happens if the buffer is NOT dissolved. Which would be better (dissolved or not dissolved) if you wanted to calculate the total flooded area in km² and why? Next, we will clip the flooded polygons to land area using the **FraserValley_studyarea.shp**.

Import **FraserValley_studyarea.shp**. Navigate to the **Clip** tool and set the **Input features** to the **flood_2m_30mBuffer** layer and the **Clip features** to the **FraserValley.shp**. Save the output as **flood_2m_30mBuffer_Clip**. Click Run.

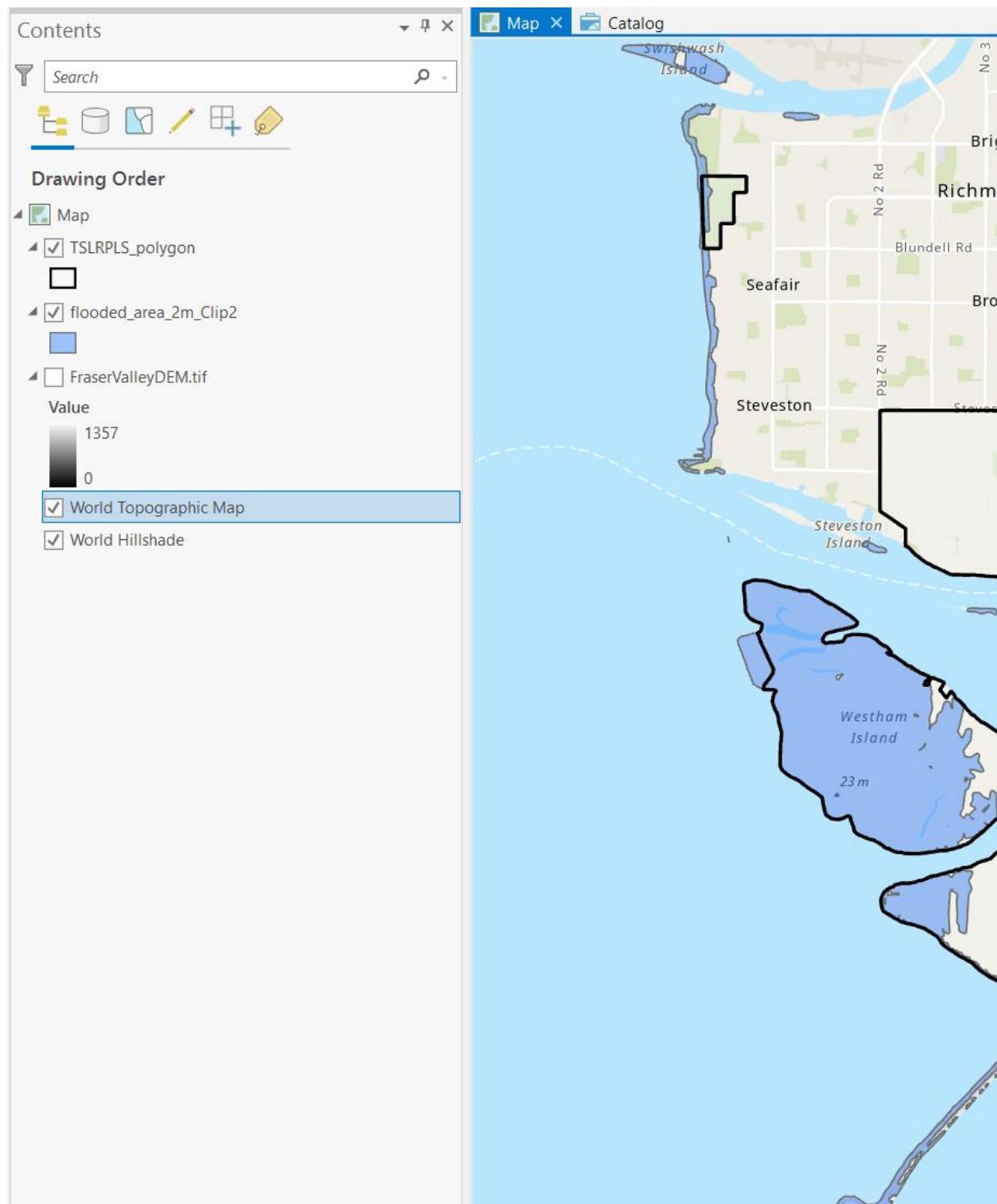
Task 2: Intersection

In this task we will **intersect** flooded areas with with land suitable for agricultural development to understand how sea level rise might impact future land use planning.

Q6: The figure below illustrates how the intersect tool works. Describe in one sentence how the output features are generated from the inputs.



Step 1: Load the **flooded_2m_30mBuffer_Clip** layer in ArcPro. Load the agricultural land shapefile, **ag_polys.shp**.



Step 2: Open the **Intersect** tool.

- Input Features: **flooded_2m_30mBuffer_Clip** and **ag_polys**
- Attributes to Join: **All attributes except feature IDs**
- Output Name: **flood_ag_intersect**

Open the attribute table of the new intersect output, and the layers you used as inputs. Notice how the attributes of both the original input layers are retained in the new intersected features.

Q7: What layer did the STATUS and AREA_SQM attributes originally come from?

	OBJECTID_1 *	Shape *	LRPLD	STATUS	FTRCD	OBJECTID	AREA_SQM	FEAT_LEN	Shape
1	1	Polygon	5116251	ALR		4527750	2545874.8692	8661.998	
2	2	Polygon	5116253	ALR		4527752	1787130.8639	10270.9852	
3	3	Polygon	5116231	ALR		4527730	496533.809	3542.4566	
4	4	Polygon	5116244	ALR		4527743	6183110.2739	10555.111	220
5	5	Polygon	5116944	ALR		4528443	1000996.4111	5602.513	2
6	6	Polygon	5119128	ALR		4530627	9900564.0578	27219.2201	12
7	7	Polygon	5119129	ALR		4530628	81454677.3214	88426.5403	4
8	8	Polygon	5118996	ALR		4530495	9844827.4152	15638.238	210
9	9	Polygon	5118997	ALR		4530496	625586.8318	3782.2811	
10	10	Polygon	5118658	ALR		4530157	2987335.2949	11147.7004	14
11	11	Polygon	5119231	ALR		4530730	587774672.3354	527781.1341	2710

Step 3: Next, we will calculate the flooded area within each agricultural polygon.

First, create a new attribute in **flood_ag_intersect**. Open the layer attribute table and click the Add button beside Field:

Fields: flooded_2m_polys_Intersect *Fields: flooded_2m...polys_Intersect

	Field:	Add	Calculate	Selection:	Select By Attributes	Zoom To	Switch	Clear	De
	OBJECTID *	Shape *	LRPLD	STATUS	FTRCD	OBJECTID	AREA_SQM	FEAT_L	
1	1	Polygon	5116251	ALR		4527750	2545874.8692	8661.	
2	2	Polygon	5116253	ALR		4527752	1787130.8639	10270.9	

In the Fields table name the new field **FloodedArea**, set the data type to **Double** and then click Save (at the top of map window).

Fields: flooded_2m_polys_Intersect *Fields: flooded_2m...polys_Intersect

	Visible	Read Only	Field Name	Alias	Data Type	Allow NULL	Highlight	Num	
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	OBJECTID	OBJECTID	Object ID	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Num	
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Shape	Shape	Geometry	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
<input checked="" type="checkbox"/>	<input type="checkbox"/>	LRPLD	LRPLD	Double	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Num	
<input checked="" type="checkbox"/>	<input type="checkbox"/>	STATUS	STATUS	Text	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
<input checked="" type="checkbox"/>	<input type="checkbox"/>	FTRCD	FTRCD	Text	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
<input checked="" type="checkbox"/>	<input type="checkbox"/>	OBJECTID_1	OBJECTID	Double	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Num	
<input checked="" type="checkbox"/>	<input type="checkbox"/>	AREA_SQM	AREA_SQM	Double	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Num	
<input checked="" type="checkbox"/>	<input type="checkbox"/>	FEAT_LEN	FEAT_LEN	Double	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Num	
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Shape_Length	Shape_Length	Double	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Num	
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Shape_Area	Shape_Area	Double	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Num	
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Int_Area	Int_Area	Double	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Num	
Click here to add a new field.									

You will now have a blank field in the attribute table.

In the attribute table, right click on the Int_Area field > Calculate Geometry. In the Property drop-down select **Area** and select **Square meters** as the Area Unit. Click OK.

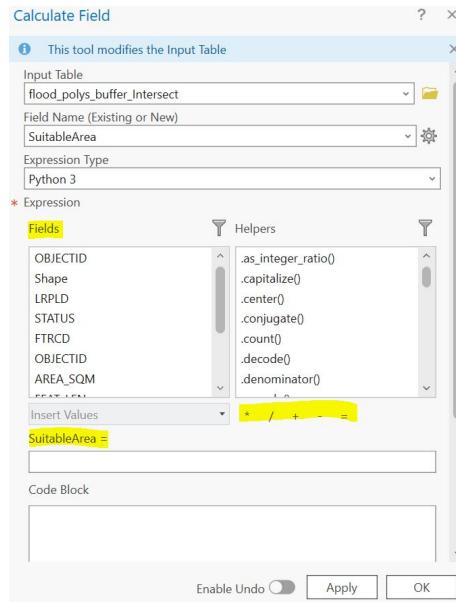
Q8: What is the total area of agricultural land impacted by flooding? Report the final answer in km² and round to two decimal places.

Q9: For the feature with LRPLD = 5118996, what percentage of the polygon is flooded? Round to two decimal places.

Task 3: Calculating & visualizing suitability

In this task, we will calculate **NON-flooded area** in each agricultural polygon to determine which regions may be most suitable to develop new agricultural land.

Step 1: Create a new attribute in **flood_ag_intersect**, set it to type = **Double** and name it **SuitableArea**. Open the attribute table and open the **Calculate Field** tool (beside the Add Field tool). Use the fields list and the math symbols to calculate **SuitableArea** as **Area of the agriculture polygon - flooded area**.



Q10: What is the LRPLD number of the polygon feature with the MOST suitable land for future agriculture? What percentage of the

total area is suitable? **Step 2:** Create another new attribute **ProportionFlooded** and use **Calculate Field** to calculate the proportion of each agricultural polygon that is flooded. Your final values should range between 0 and 1. Make sure the data type for the attribute is **Double** so it can store decimal values.

Q11: What expression did you use to calculate ProportionFlooded? (Copy it from the Calculate Field tool). **Step 3:** Next, we will **standardize** the SuitableArea attribute so its values range from 0 to 1 (the same range as the ProportionFlooded).

Create a new attribute **SuitArea_Std** and use the following equation to calculate it:

$$\text{SuitArea_Std} = (\text{SuitableArea} - \text{Minimum}) / (\text{Maximum} - \text{Minimum})$$

Where Minimum is the minimum value of SuitableArea, and Maximum is the Maximum value of suitableArea.

Step 4: Next, we will create a **Suitability Score** that combines the **ProportionFlooded** attribute and the **SuitArea_Std** attribute. Create a new attribute to represent a final **Suitability Score** and set the Data Type to **Double**.

Use the equation below to create a Suitability Score based on the proportion of non-flooded land and the total suitable area (standardized). The final values will range from 0 (least suitable) to 1 (most suitable):

$$((1-\text{ProportionFlooded}) + \text{SuitArea_Std})/2$$

Q12: What is the average suitability score? **Step 5:** Next, we will **join** the new attributes from the **flood_ag_intersect** layer to the original **ag_polys** shapefile so that we can visualize the Suitability Score within the agricultural polygons.

Right click on the **ag_polys** layer in the Contents pane > Joins and Relates > Add Join.

- Join Features: **flood_ag_intersect**
- Join Attribute: **LRPLD**
- **Un-check** Keep all target features

Step 6 Change the Symbology of the **ag_polys** layer to reflect the **Suitability Score**. Finally, create a map with the following elements and include it in your final deliverables:

- Agricultural polygons color coded by suitability score
- Title

- North arrow
- Scale bar
- Legend

Q13: Based on your analysis where is the best areas to safely expand agricultural land? What other land use types may compete for this land?