

ESRM433/SEFS533

Lab 2

Objectives:

- Intro to Coordinate Reference Systems (CRS)
- Lidar Metadata
- Introduction to LidR and RStudio
- Accessing online help and documentations for LidR and RStudio

Data and Software:

- Watch this video about Coordinate Reference Systems:
 - <https://www.youtube.com/watch?v=xJyJKbZFic>
- Washington State Lidar Portal <http://lidarportal.dnr.wa.gov/>
 - Download data and store on Madrona Desktop at **ESRM433\LAB2**
 - Network directory name will be:
 - `//fs-persona.sefs.uw.edu/student_redirect$/*YourNetID*/Desktop/ESRM433/LAB2`
- RStudio. You can use via Madrona or download on your own computer here: <https://rstudio.com/products/rstudio/download/>
 - RStudio requires an installation of R as well: <https://cran.rstudio.com/>
 - A basic tutorial about using RStudio
 - <https://education.rstudio.com/learn/beginner/>
- lidR R package for lidar data manipulation <https://github.com/Jean-Romain/lidR>
 - lidR tutorials to supplement this lab: <https://github.com/Jean-Romain/lidR/wiki>

What you will turn in:

- Submission Template in PDF file format submitted via Canvas
-

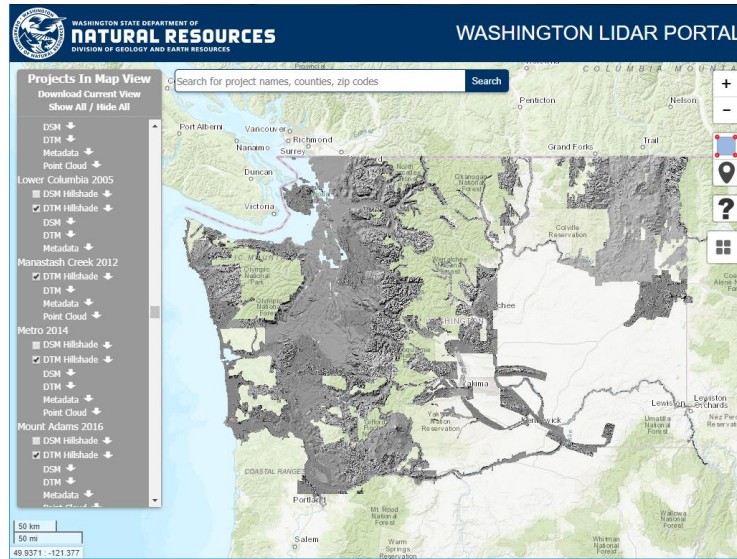
Welcome to Lab 2 for ESRM433/SEFS533!

The goals of this lab are to 1) introduce you to the information contained within lidar metadata pdfs
2) Introduce you to LidR, the R package that we will be using for much of this class and and show you where you can find online documentation about LidR and Rstudio to help you if you get stuck.

PART 1 Lidar Metadata:

You will navigate around the Washington State DNR Lidar Portal and explore some of the metadata from various lidar projects.

In lab 1 you were tasked with downloading a single .laz tile of a lidar acquisition. We are now going to more closely examine the data available within the portal and explain some of the terms being used.



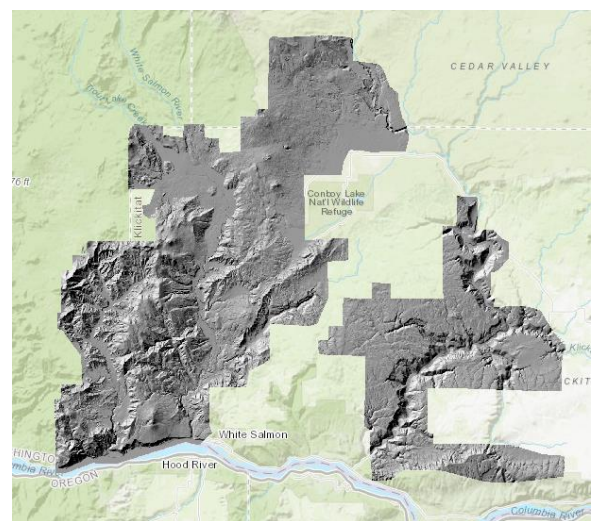
There are many projects (also referred to as acquisitions) that have data available for download. The sizes of these acquisitions vary dramatically, and a single acquisition can be broken up into several, non-contiguous areas. All acquisitions are made up of tiles of data and an entire project's data size can be in excess of 200GB. Some areas have multiple acquisitions covering them. Using the lidar portal, you are able to toggle off and on DTM polygons for each project as well as search for acquisitions by name or date.

IMPORTANT TO REMEMBER, the Washington DNR converts the projections of all lidar acquisitions to the coordinate reference system (CRS): Washington State Plane South, Datum: NAD83, Units: feet, when they are added to the lidar portal. The coordinate reference system reported in the metadata pdf is what the vendor supplied the DNR, and not the projection of the downloaded data from the DNR lidar portal. We will be talking more about CRSs next lab. In the meantime, watch this video, it will help a lot!

<https://www.youtube.com/watch?v=xJyJIKbZFfc>



Cold Springs 2008 acquisition, a relatively small project area.




Klickitat 2015 acquisition, covering some of the same area as Cold Springs but is much larger with two non-contiguous areas.

Lab 2



You can use the search bar to look for acquisitions based on names, counties, or zip codes. The names of the acquisitions include the year the data was collected enabling you to search for all acquisitions made in a specific year. Note that the “Projects In Map View” only lists the projects that are within the current extent of your web browser window. You can also change the base map from

topography to imagery 

QUESTION 1: What year are the oldest projects from (just start searching years working your way back)?

QUESTION 2: What year are the most recent projects? If there aren't any from the current year, why the delay?

We are now going to explore some of the data more closely. We are going to download a single tile from the Pack Forest 2013 acquisition as well as the meta data for the entire acquisition.

Go to: <https://lidarportal.dnr.wa.gov/#46.84264;-122.31453;18>

Those are coordinates and the zoom level for the DNR site.

ESRM433/SEFS533

Lab 2

If you “Hide All” for the Projects in Map View, and turn on an imagery basemap, you may recognize the Pack Forest buildings. You want to use the **PIN TOOL** and drop it in the same location as the image above. Click “Deselect All” and then click the Metadata and Point Cloud boxes. The entire Pack Forest acquisition is many gigabytes but a single tile and the metadata pdf is only ~44MB.

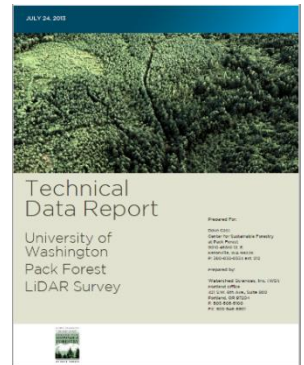
MAKE SURE YOU ARE ONLY DOWNLOADING ~44MB

Download data to **Madrona Desktop\ESRM433\LAB2**

//fs-persona.sefs.uw.edu/student_redirect\$/*YourNetID*/Desktop/ESRM433/LAB2

Unzip and you should have a tile of ALS data and a PDF.

Take a moment to look through the PDF and answer the following questions.
There is a glossary of terms at the end of this lab.



QUESTION 3: What lidar project did you look at? What company collected the ALS and when was the data acquired?

QUESTION 4: What does the technical report say the map datum, projection, and units (feet or meters) are?

QUESTION 5: What is the map datum, projection, and units (feet or meters) of the data when downloaded from the Washington Lidar Portal?

QUESTION 6: How many acres were within the project? What is AOI and TAF?

QUESTION 7: What scanner was used to collect the data?

QUESTION 8: What was the targeted point density?

QUESTION 9: Give one statistic about the accuracy of the lidar.

QUESTION 10: What was used for Ground Control (Ground Check Points)?

QUESTION 11: What is one element, concept, or term in the report that you aren't that familiar with?

QUESTION 12: INCLUDE A SCREEN SHOT OF ONE OF THE COOL IMAGES WITHIN THE REPORT AND CAPTION IT IN YOUR OWN WORDS.

QUESTION 13: Lidar wasn't the only data collected during the flight. What other remote sensing data was collected?

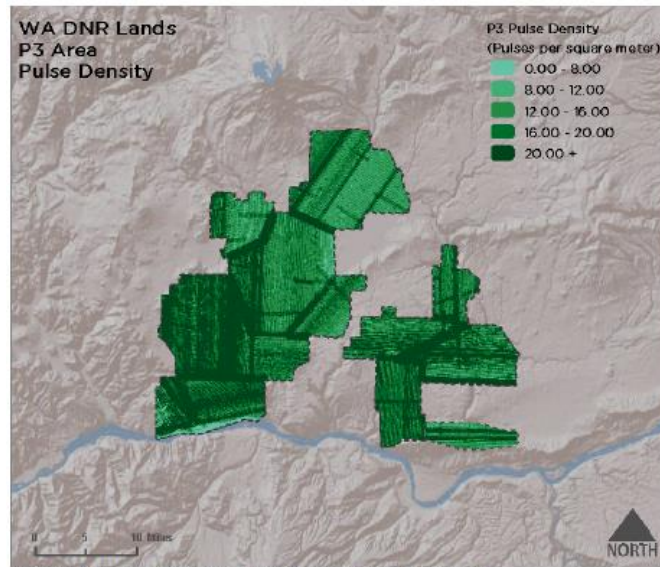


Figure 13: Map of first return densities per 30 m X 30 m cell

This is a figure from the Klickitat 2015 metadata. For your quiz, caption a figure with your own words (a few sentences)

PART 2 Rstudio and LidR

For the majority of this class, we are going to be using the R package LidR to work with the lidar data. You don't need to have a strong background in R to be successful, as all the steps will be laid out for you with all the inputs clarified. To run though some very general Rstudio tutorials go to:

- o <https://education.rstudio.com/learn/beginner/>

IMPORTANT: Make sure you use good file structures and data organization. Always make sure you know what directory files are in and where you are saving files.

These labs will be assuming two things:

- You are using Madrona to complete the labs (all programs used in this class are free and you are encouraged to download the programs and work on your own computer but that potentially limits our ability to trouble shoot with you)
- That you are storing your data in your Madrona desktop in a folder "ESRM433"
 - Thus my lab two data on my Madrona instance would be located at
 - `//fs-persona.sefs.uw.edu/student_redirect$/JonBatch/Desktop/ESRM433/LAB2`
 - i. The highlighted area will be your own uw net id
 - You can name it SEFS533 if you want

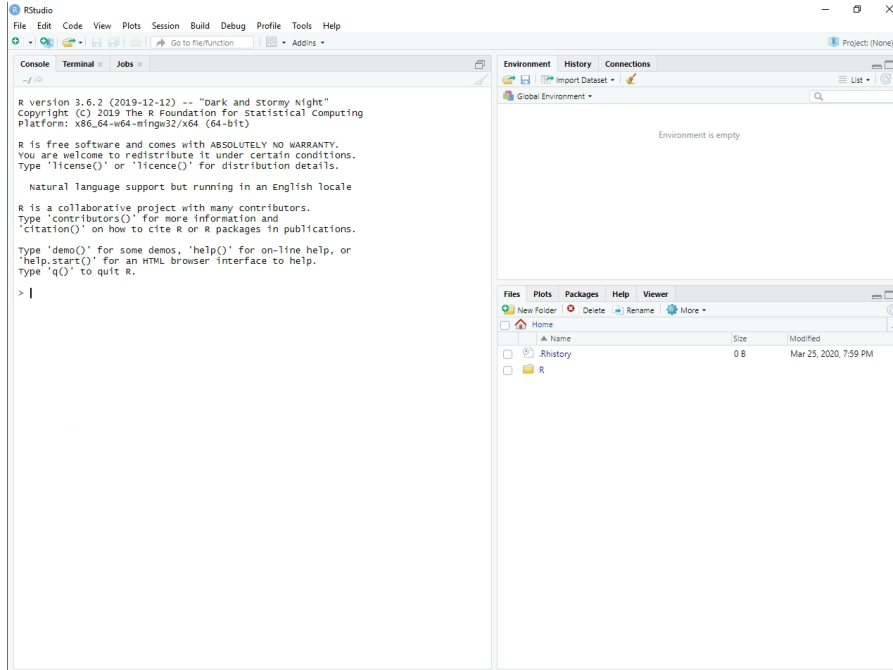
IMPORTANT: Keep track of how much data you are downloading and storing and WHERE it is all being stored. We will be dealing with large data sets and if you dump several gigs of data in your Madrona account, you may get locked out. Delete all unused data and don't download duplicate datasets.

ESRM433/SEFS533

Lab 2

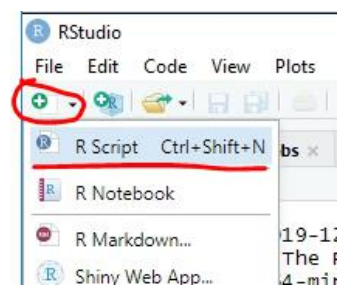
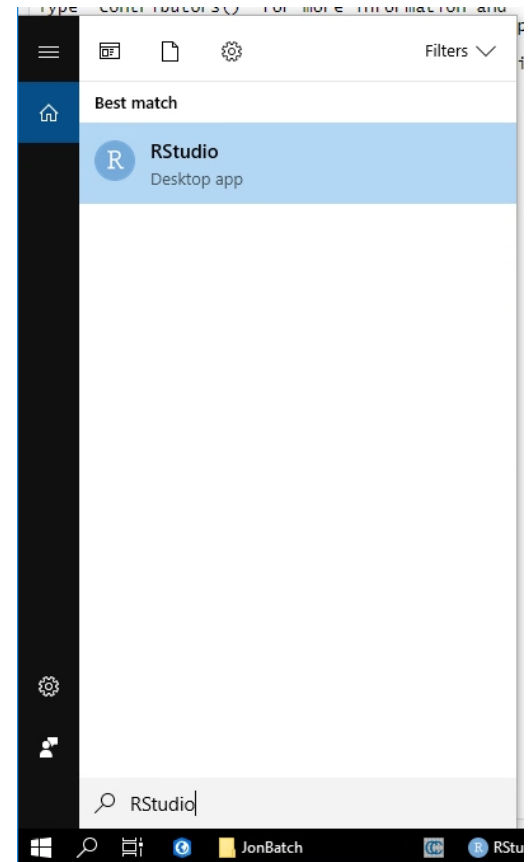
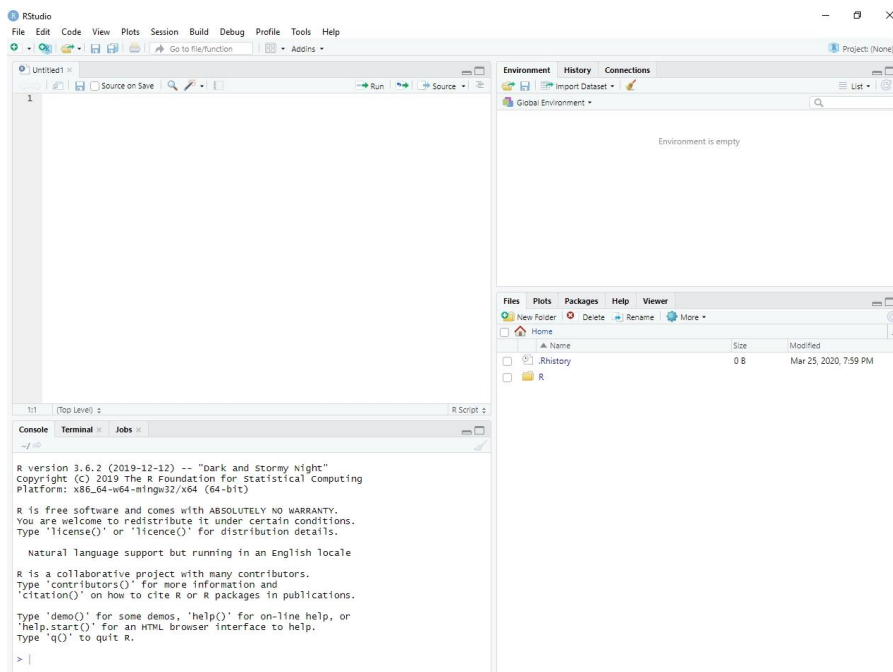
Step 1: You can launch RStudio from inside Madrona by typing RStudio in the search bar. I would suggest pinning it to your taskbar.

This should launch RStudio and hopefully you will get a screen like this:



First thing will want to do is to create a new R Script

This will open a new box within the R Studio Window:



ESRM433/SEFS533

Lab 2

Assuming you have a directory with your downloaded laz tile located at:

//fs-persona.sefs.uw.edu/student_redirect\$/*YourNetID*/Desktop/ESRM433/LAB2

You are going to want to point RStudio to your directory where all your data is stored. This will make future steps easier.

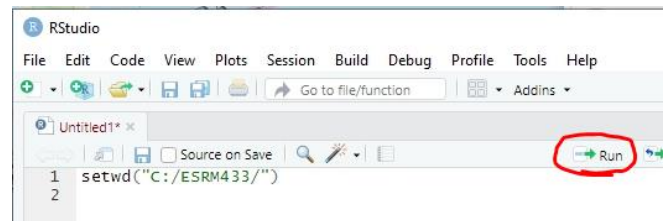
In your new R Script window type:

```
in
```

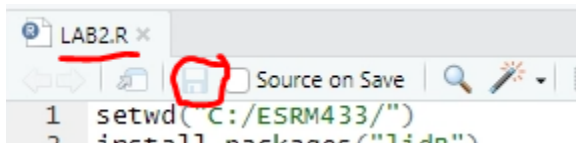
`setwd` This is 'set working directory' which just tells R to look at this location for files
(`"`) This defines what directory to look at. Important thing to notice is that the slashes are in the different direction that the slashes from the windows commands. R is very particular about slash direction, spaces, and capitalization.

Note that the screen shots use "C:/ESRM433/". You can use this directory only if you are using your own computer. If you are using Madrona you must use the network address given above. The screenshots are incorrect if using Madrona.

From the R script window, to run a line of code, place your cursor on the line and press **Ctrl+Enter** or you can press the 'Run' button, but your cursor must be on the line you want to run



Also feel free to go ahead and save your new R script as Lab2 in your Lab2 folder.



Next we are going to install the lidR package. Type:

```
install.packages("lidR")
```

`install.packages` This tells RStudio you want to install a set of specialized scripts that are called packages
(`"lidR"`) This is the name of a package that is listed in an online RStudio directory that can be installed locally

Click yes or OK on any pop ups that ask about downloading additional files. This will take a little bit.

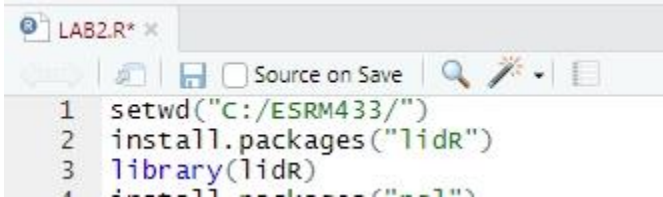
Once all the packages are downloaded and installed, you still have to tell R Studio to use lidR in the script you are producing. Type:

```
library(lidR)
```

`library` loads the specified package

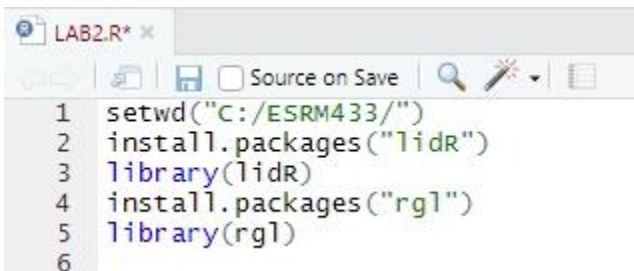
ESRM433/SEFS533

Lab 2



```
1 setwd("C:/ESRM433/")
2 install.packages("lidR")
3 library(lidR)
```

You need to install and load one more package to visualize lidar point clouds. The package is called `rgl`. Install the package and use `library` to load it.



```
1 setwd("C:/ESRM433/")
2 install.packages("lidR")
3 library(lidR)
4 install.packages("rgl")
5 library(rgl)
```

OK, we are now ready to start using R to look at some lidar data. Assuming that your working directory is Madrona Desktop /ESRM433, and your downloaded lidar tile is in a lab2 folder, and is named `wa_031_rp.laz`... type:

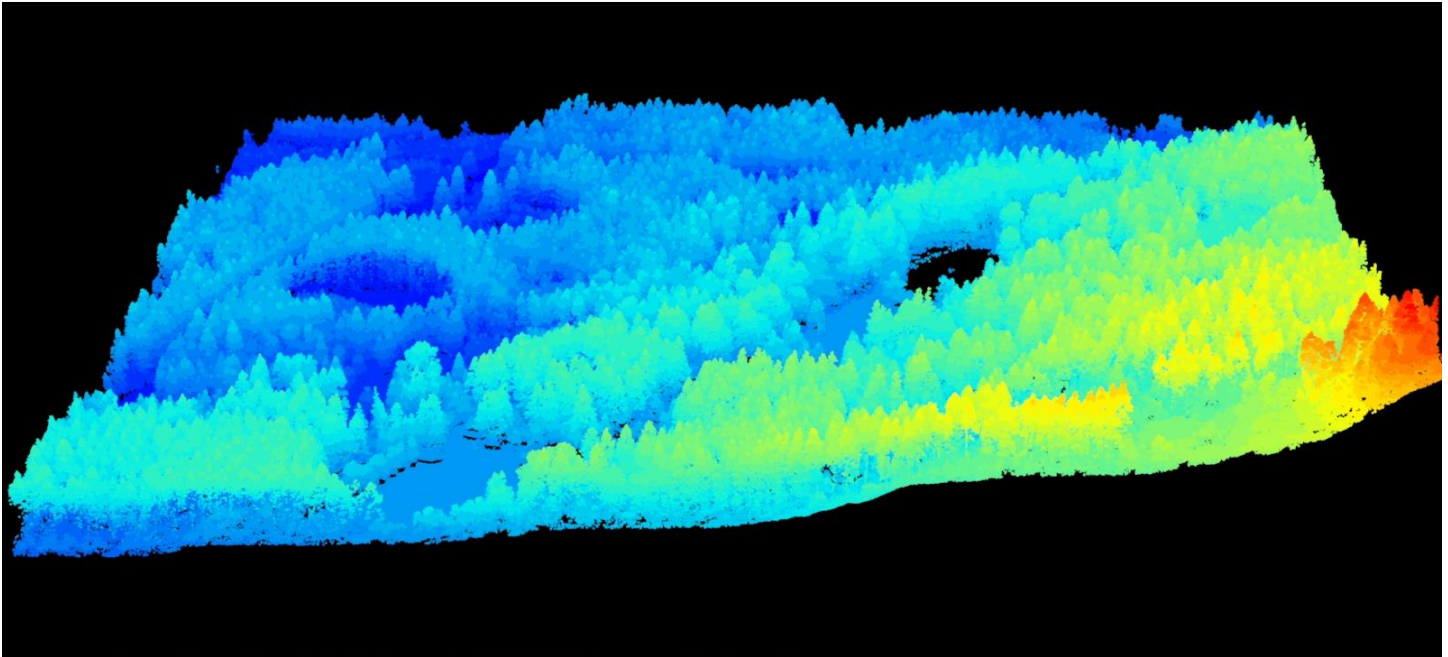
```
LASfile <- ("Lab2/wa_031_rp.laz")
las <- readLAS(LASfile)
plot(las)
```

`LASfile <- ("Lab2/wa_031_rp.laz")` This sets the term "LASfile" to represent the file `wa_031_rp.laz`. The `<-` symbol denotes a name as being representative of a file, or function.

`las <- readLAS(LASfile)` This sets the term "las" as representing the function "readLAS" applied to the file "LASfile". `readLAS` is a function of the `lidR` package which allows R to understand the file structure of `las` and `laz` files.

`plot(las)` Commands RStudio to visualize `las`. This may take a moment to load.

```
1 install.packages("lidR")
2 install.packages("rgl")
3 library(lidR)
4 library(rgl)
5
6 setwd("//fs-persona.sefs.uw.edu/student_redirect$/jonbatch/Desktop/ESRM433/")
7
8 LASfile <- ("Lab2/wa_031_rp.laz")
9 las <- readLAS(LASfile)
10 plot(las)
```

QUESTION 14: Zoom into an interesting feature in the point cloud and submit a high quality screenshot of the feature and describe what it is.

OK, so we've visualized a point cloud using RStudio and lidR. But we could already do that with an easier program with better visualization capability, CloudCompare. Why do we need lidR? lidR has a lot of capacity to derive metrics and manipulate lidar point clouds that cloudcompare isn't able to replicate. We will be using both in future labs depending on what we need to accomplish. For now, let's continue exploring lidR. Type:

```
print(las)
```

`print(las)` print is a generic R command, in this case when run on a realLAS function, it prints out basic information about the las file.

QUESTION 15: What is the size of the file (memory)? What is the extent, and what coordinate system are those numbers in? What is the area covered by the ALS tile? How many points?

ESRM433/SEFS533

Lab 2

The names field hopefully looks familiar to you. These are the scalar fields of a las file, with a few additional user defined fields. For more information about las file format check out: [LAS specification](#) version 1.4.

| Item | Format | Size | Required |
|--|---------------------|---------|----------|
| X | long | 4 bytes | * |
| Y | long | 4 bytes | * |
| Z | long | 4 bytes | * |
| Intensity | unsigned short | 2 bytes | |
| Return Number | 3 bits (bits 0 – 2) | 3 bits | * |
| Number of Returns (given pulse) | 3 bits (bits 3 – 5) | 3 bits | * |
| Scan Direction Flag | 1 bit (bit 6) | 1 bit | * |
| Edge of Flight Line | 1 bit (bit 7) | 1 bit | * |
| Classification | unsigned char | 1 byte | * |
| Scan Angle Rank (-90 to +90) – Left side | char | 1 byte | * |
| User Data | unsigned char | 1 byte | |
| Point Source ID | unsigned short | 2 bytes | * |
| GPS Time | double | 8 bytes | * |

You can get additional information about the las file with the command `print(las@header)`

Go ahead and try it and see what additional information is available.

Next, we are going to run a check on our las file to see if the data conforms to las specifications, to see if there are any erroneous points, and to check if any processing has been done on the file.

`Las_check(las)`

```
> lascheck(las)

Checking the data
- Checking coordinates... ✓
- Checking coordinates type... ✓
- Checking attributes type... ✓
- Checking ReturnNumber validity... ✓
- Checking NumberOfReturns validity... ✓
- Checking ReturnNumber vs. NumberOfReturns... ✓
- Checking RGB validity... ✓
- Checking absence of NAs... ✓
- Checking duplicated points...
  Δ 53 points are duplicated and share XYZ coordinates with other points
- Checking degenerated ground points... ✓
- Checking attribute population...
  Δ 'EdgeOfFlightline' attribute is not populated.
- Checking flag attributes... ✓
Checking the header
- Checking header completeness... ✓
- Checking scale factor validity... ✓
- Checking point data format ID validity... ✓
- Checking extra bytes attributes validity... ✓
- Checking coordinate reference system... ✓
Checking header vs data adequacy
- Checking attributes vs. point format... ✓
- Checking header bbox vs. actual content... ✓
- Checking header number of points vs. actual content... ✓
- Checking header return number vs. actual content... ✓
Checking preprocessing already done
- Checking ground classification... yes
- Checking normalization... no
- Checking negative outliers... ✓
- Checking flightline classification... yes
> |
```

Most of this information won't be that useful for you, but pay attention to the "Checking preprocessing

ESRM433/SEFS533

Lab 2

already done" section. Here you can see that ground classification has been done on this las file and that normalization hasn't been done. We will be coming back to these in future labs.

Last Step!

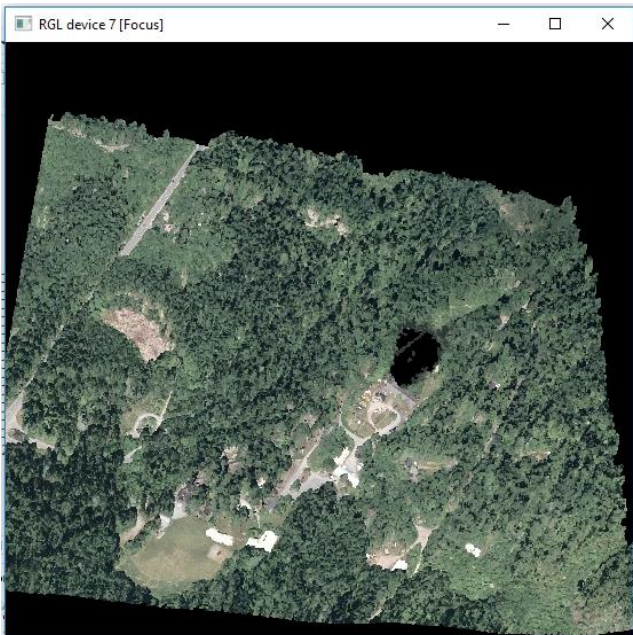
Lets see a few other ways that lidR can visualize point clouds. Going back to the plot function, type

```
plot(las, color="Intensity")  
  
plot(las, color="Intensity", trim = 100)
```

The original plot function colored the points based on height. You can color them based on the intensity of the return and you can trim the color ramp if there are a few outliers that are causing issues with the color ramp values being displayed.

Lastly, type

```
plot(las, color="RGB")
```



QUESTION 16: Lidar records the position of an object that reflects back sufficient laser energy to record a point, and how much of the energy of the original pulse was reflected back. How is it possible that this ALS point cloud data is full color? (answer is in the metadata pdf)

QUESTION 17: Zoom into the RGB full color ALS point cloud and submit a screenshot of some cool feature. Try using cloud compare as it has better visualization abilities.

GRADUATE STUDENT

Repeat questions 14 & 15 as well as a screenshot of your lascheck output (like the image included in this lab) for a different tile. Does your tile have RGB values? Include a map of the location you picked and make sure your tile you download is less than 50MB.

Glossary

1-sigma (σ) Absolute Deviation: Value for which the data are within one standard deviation (approximately 68th percentile) of a normally distributed data set.

1.96 * RMSE Absolute Deviation: Value for which the data are within two standard deviations (approximately 95th percentile) of a normally distributed data set, based on the FGDC standards for Fundamental Vertical Accuracy (FVA) reporting.

Accuracy: The statistical comparison between known (surveyed) points and laser points. Typically measured as the standard deviation (sigma σ) and root mean square error (RMSE).

Absolute Accuracy: The vertical accuracy of LiDAR data is described as the mean and standard deviation (sigma σ) of divergence of LiDAR point coordinates from ground survey point coordinates.

To provide a sense of the model predictive power of the dataset, the root mean square error (RMSE) for vertical accuracy is also provided. These statistics assume the error distributions for x, y, and z are normally distributed, and thus the skew and kurtosis of distributions are also considered when evaluating error statistics.

Relative Accuracy: Relative accuracy refers to the internal consistency of the data set (i.e., the ability to place a laser point in the same location over multiple flight lines), GPS conditions, and aircraft attitudes. Affected by system attitude offsets, scale, and GPS/IMU drift, internal consistency is measured as the divergence between points from different flight lines within an overlapping area. Divergence is most apparent when flight lines are opposing. When the LiDAR system is well calibrated, the line-to-line divergence is low (<10 cm).

Root Mean Square Error (RMSE): A statistic used to approximate the difference between real-world points and the LiDAR points. It is calculated by squaring all the values, then taking the average of the squares and taking the square root of the average.

Data Density: A common measure of LiDAR resolution, measured as points per square meter.

Digital Elevation Model (DEM): File or database made from surveyed points, containing elevation points over a contiguous area. Digital terrain models (DTM) and digital surface models (DSM) are types of DEMs. DTMs consist solely of the bare earth surface (ground points), while DSMs include information about all surfaces, including vegetation and man-made structures.

Intensity Values: The peak power ratio of the laser return to the emitted laser, calculated as a function of surface reflectivity.

Nadir: A single point or locus of points on the surface of the earth directly below a sensor as it progresses along its flight line.

Overlap: The area shared between flight lines, typically measured in percent. 100% overlap is essential to ensure complete coverage and reduce laser shadows.

Lab 2

Pulse Rate (PR): The rate at which laser pulses are emitted from the sensor; typically measured in thousands of pulses per second (kHz).

Pulse Returns: For every laser pulse emitted, the number of wave forms (i.e., echos) reflected back to the sensor. Portions of the wave form that return first are the highest element in multi-tiered surfaces such as vegetation. Portions of the wave form that return last are the lowest element in multi-tiered surfaces.

Real-Time Kinematic (RTK) Survey: A type of surveying conducted with a GPS base station deployed over a known monument with a radio connection to a GPS rover. Both the base station and rover receive differential GPS data and the baseline correction is solved between the two. This type of ground survey is accurate to 1.5 cm or less.

Post-Processed Kinematic (PPK) Survey: GPS surveying is conducted with a GPS rover collecting concurrently with a GPS base station set up over a known monument. Differential corrections and precisions for the GNSS baselines are computed and applied after the fact during processing. This type of ground survey is accurate to 1.5 cm or less.

Scan Angle: The angle from nadir to the edge of the scan, measured in degrees. Laser point accuracy typically decreases as scan angles increase.

Native LiDAR Density: The number of pulses emitted by the LiDAR system, commonly expressed as pulses per square meter.

R Appendix

```
1 install.packages("lidR")
2 install.packages("rgl")
3 library(lidR)
4 library(rgl)
5
6 setwd("//fs-persona.sefs.uw.edu/student_redirect$/jonbatch/Desktop/ESRM433/")
7
8 LASfile <- ("Lab2/WA_031_rp.laz")
9 las <- readLAS(LASfile)
10 plot(las)
11 print(las)
12 print(las@header)
13 las_check(las)
14
15 plot(las, color="Intensity",trip=100)
16 plot(las, color="RGB")
```
