Outline of presentation

Create dataset with a function that inputs the top left corner and iteratively populates a dataframe with points ~ 10 m apart

Read in a file with elevatr::get\_elev\_point

* Point out proj, and spend ~5 min discussing the importance of projections
  + Draw a sphere on the board! ☺ Unravel to reveal projection
  + Datum and projection
    - Check to see if your projection exists using **is.projected()**
    - Geographic CS vs. Projection CS
      * Geo (Lat, Long) and spherical, Proj (X, Y) and 2-d
      * semi-minor and major axis comparison for x and y max depending on Geoid, ellipsoid, or spheroid
      * datum is the elevation from the spheroid to the earth’s surface
        + local variation in elevation
* Describe importance of defining the source as a default … or not
  + Working across the globe
    - Particularly if working across different latitudes, where the equatorial bulge may impact a substantial variation
  + Show where the default is defined just to review projection and datum options
* Explain difference between using this function vs. elevatr::get\_eleve\_raster
  + If you don’t have elevation information, this function takes lat and long coordinates and adds an additional variable to the dataset—elevation
  + Alternatively, instead of outputting a dataset with additional information, get\_elev\_raster can input either datasets OR a raster and outputs the information as a raster

Call in external raster

Read in file with elevatr::get\_eleve\_raster

* When is it advantageous to use rasters instead of dataframes??
  + Location (available inputs)-dependent OR output dependent
* Zoom level—what are you looking for?

**Extra’s/Spicy Ideas**

For the preliminary libraries, maybe consider including ~~ggplot,~~ ~~patchwork,~~ ~~and dplyr?? (whichever one contains melt) to point out necessary cleaning from NASA/NOAA files since these would be the standard servers beyond Mapzen, Amazon Web Services, USGS, and epqs, particularly for remotely sensed~~

Use patchwork with different zoom levels!!

Also, is there an additional package that can overlay this information onto other types of data that I can include as well for a fun grand finale?

* *Fasterize to show shape file to raster*
* **Raster::flowPath and ::density**
* AdehabitatMA::mimage and ::morphology
* sp::SpatialPoints

Two potential examples—using data from online or example options

Create dataset by setting a max and min value to randomize lat and long values

* set seed?
* Allow everyone to set their own max and min based on the location of interest
* 1000 data points

use the enchantments as the example—make a joke about the shitty map I used to hike around there and include a photo of my hike on the r markdown—show the unique water features with flowpath – when choosing a unique region of the US, suggest a unique drainage system!

~~Could also run a query on max and min heights?~~

~~Array of coordinates for summits to compare~~? Use summary!!

When plotting, specify zoom levels (trade off between resolution and processing time)

Add contour lines to the plot

**Class "CRS" Of Coordinate Reference System Arguments**

Interface class to the PROJ.4 projection system. The class is defined as an empty stub accepting value NA in the sp package. If the rgdal package is available, then the class will permit spatial data to be associated with coordinate reference systems. The arguments must be entered exactly as in the PROJ.4 documentation, in particular there cannot be any white space in +<arg>=<value> strings, and successive such strings can only be separated by blanks. Note that only **“+proj=longlat +ellps=WGS84”** is accepted for geographical coordinates, which must be ordered (eastings, northings); the “+ellps=” definition must be given (or expanded internally from a given “+datum=” value) for recent versions of the PROJ.4 library, and should be set to an appropriate value.

EPSG codes for commonly used CRS (in the U.S.) Latitude/Longitude WGS84 (EPSG: 4326) ## Commonly used by organizations that provide GIS data for the entire globe or many countries. CRS used by Google Earth

PROVIDE WEBSITE TO FIND APPLICABLE APSG CODE FOR PROJECTION

<http://www.epsg-registry.org/>

# Plot distribution of raster values

DSMhist<-hist(DSM\_HARV,

breaks=3,

main="Histogram Digital Surface Model\n NEON Harvard Forest Field Site",

col="wheat3", # changes bin color

xlab= "Elevation (m)") # label the x-axis

## Warning in .hist1(x, maxpixels = maxpixels, main = main, plot = plot, ...):

## 4% of the raster cells were used. 100000 values used.

Can also plot a histogram

Will need the following links:

<https://aws.amazon.com/public-datasets/terrain/>

<http://www.epsg-registry.org/>