**GIS and remote sensing for conservation and evolutionary biology**

**Session 3: computer lab exercise**

**Aim:** To locate and view Landsat images and to produce an NDVI image.

**Part 1: Locate a Landsat Scene in Earth Explorer**

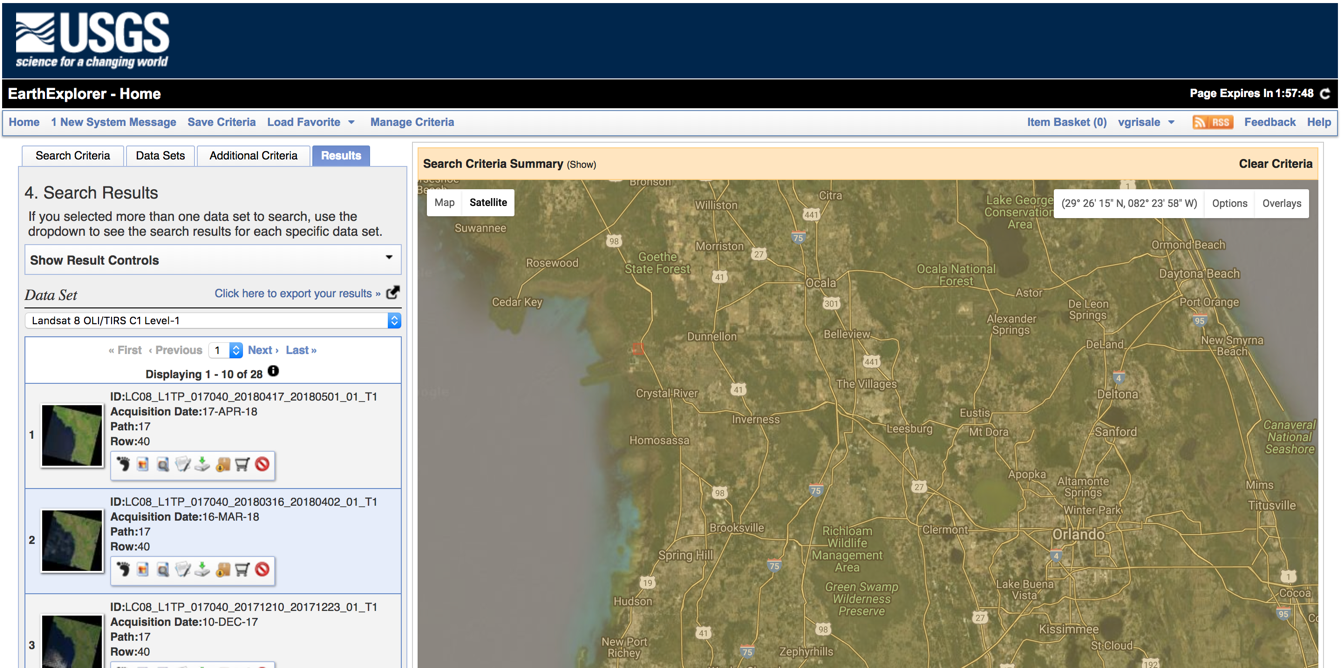
All archived Landsat scenes are now available for free download from the USGS. There are USGS tools commonly used for finding and downloading Landsat, commonly Earth Explorer. First, go to Earth Explorer and register to create a free account. <http://earthexplorer.usgs.gov/>

Go to the Earth Explorer website and zoom to an area of interest within the climate extent created last session. The **Search Criteria** tab is made up of three panes to help narrow your search: 1) adding address or shapefile data, 2) identifying areas by defining geometry, and 3) selecting data from a range of dates. In the second pane, click the **Polygon** tab, then click *Use Map*. This will tell the website that you only want the imagery from this viewing extent. Lastly, set a date range using the past two to four years (you may need to widen the range based on available data). Under **Data Sets**, go to *Landsat > Landsat Collection 1 Level-1* to see available data. Landsat7 are scenes collected after a sensor failure in 2003. These scenes are missing data stripes (scan lines), which can be difficult to deal with. I recommend that you select a Landsat 8 product; anything that uses TIRS (Thermal Infrared Sensor).

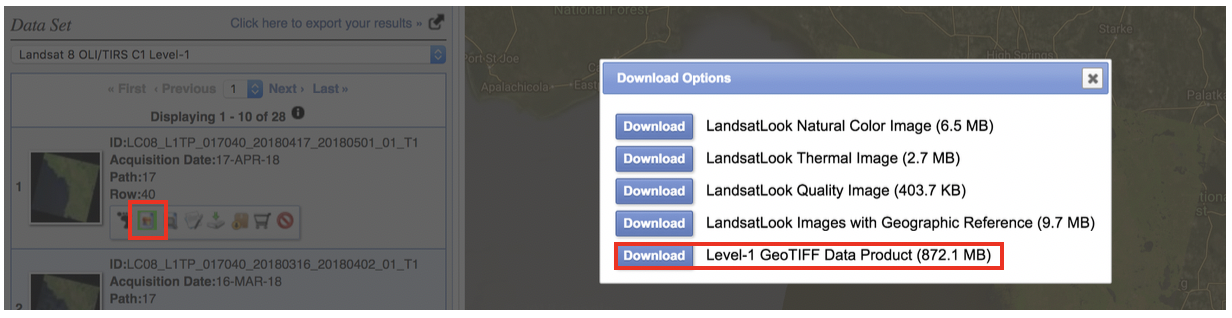
**Example of Landsat 7, SLC-off**



Ideally, you will want to download tiles with low cloud cover. Earth Explorer offers two types of cloud cover measurements: Land Cloud Cover, which disregards clouds over oceans, and Scene Cloud Cover, measuring all clouds within the view. You may need to determine which cloud cover measurement has more tiles available for your species. Decide whether you are more interested in Land Cloud Cover, or Scene Cloud Cover. Under **Additional Criteria**, specify that you want images with less than 10% cloud cover (depending on the data available in the cloud cover scenario you choose, you may need to relax this number).



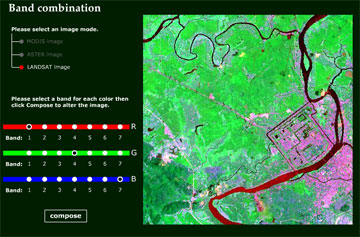
Click **Results**, and you will see many tiles with options. To see each tile, click on “Show Browse Overlay”(second button from the left; mouse over to see label). When you find a tile with few clouds and good coverage, click **download options** then Download **Level 1 GeoTIFF Data Product**.



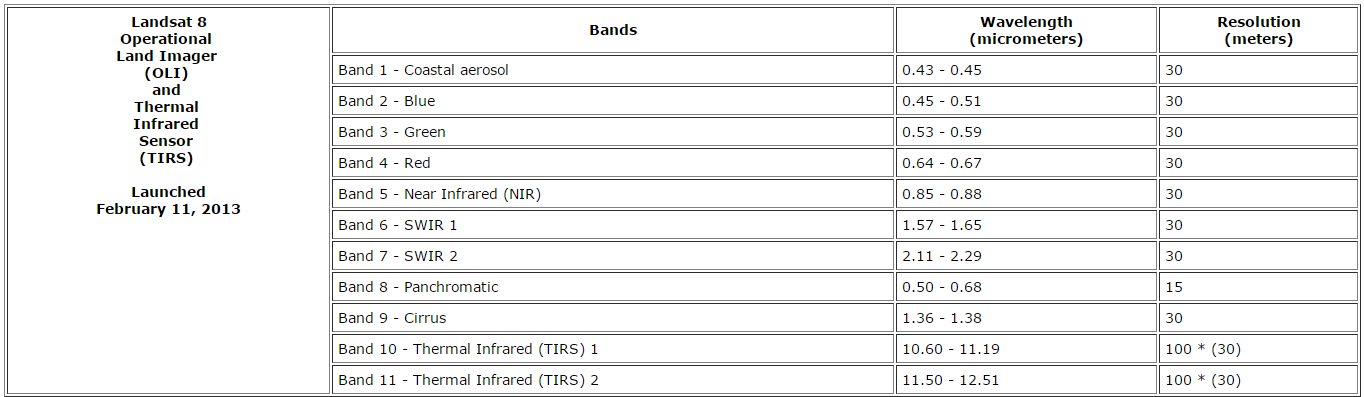
**Part 2: Creating a multi-band GeoTIFF from individual files using QGIS**

While your data is downloading, familiarize yourself with Landsat bands using the Band combination interactive on the Biodiversity Informatics Facility website. <http://biodiversityinformatics.amnh.org/interactives/bandcombination.php>.

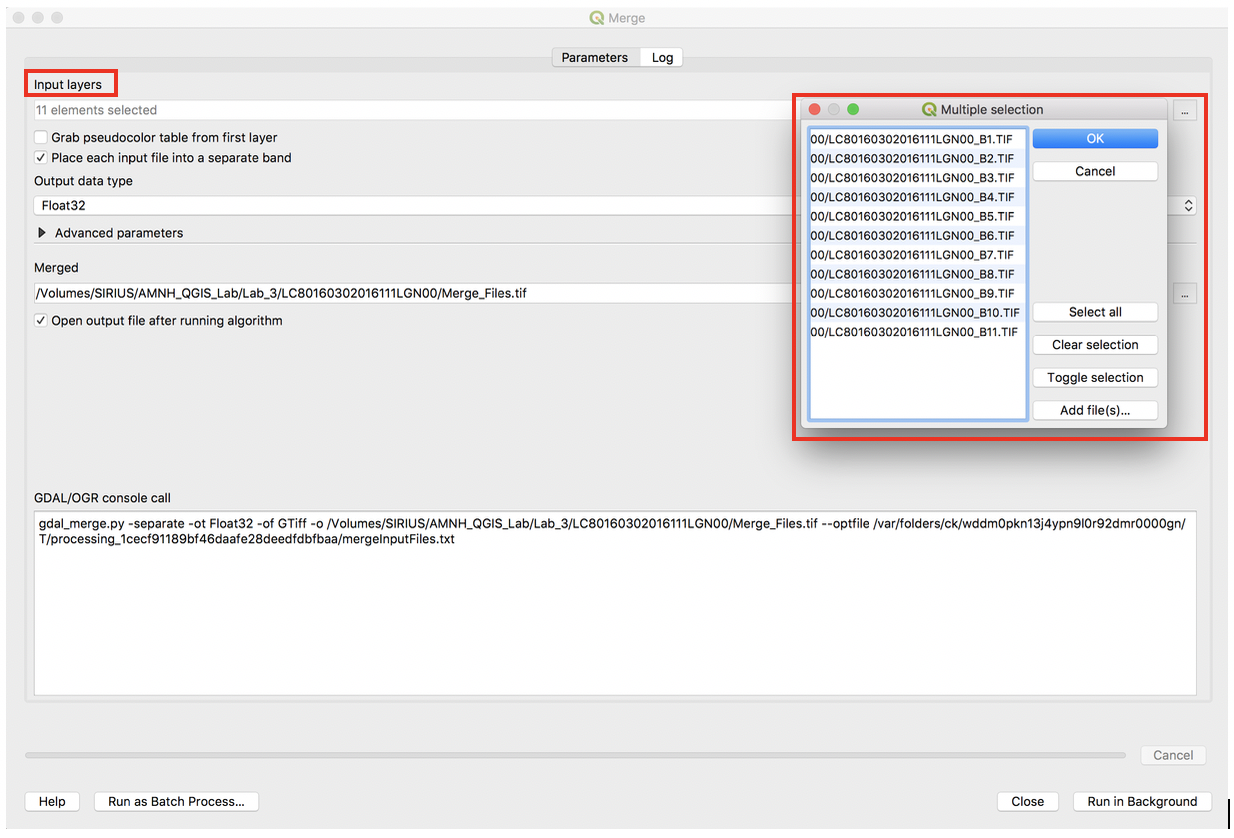
1. Select LANDSAT image from the image mode choices. Experiment with band combinations discussed during the lecture. Try Red = 3, Green = 2 and Blue = 1 for a natural color view (hit compose). Then try 4,3,2 for false color and the 4,5,3 and 7,4,2 color combinations.



1. When the data has completed downloading, place the file in your project directory on your computer.
2. You will need to unpack the tar.gz file (In windows you can use “7-zip”, Mac users can use “archive utility”; or this is easily done in R using the function untar()). This will give a list of rasters and text files. The rasters represent the various bands available. The bands are named according to this table.



1. Now we need to merge all these layers into one composite raster with 11 bands. Go to **Raster > Miscellaneous > Merge.** Under *Input layers*, click on the ellipses, then *Add Files* to navigate to your raster bands. Select all bands as input files (i.e. 1-11). Then name an output file. Click the box for **Place each input file into a separate band**. **Run** the merge, and **Close** when finished.



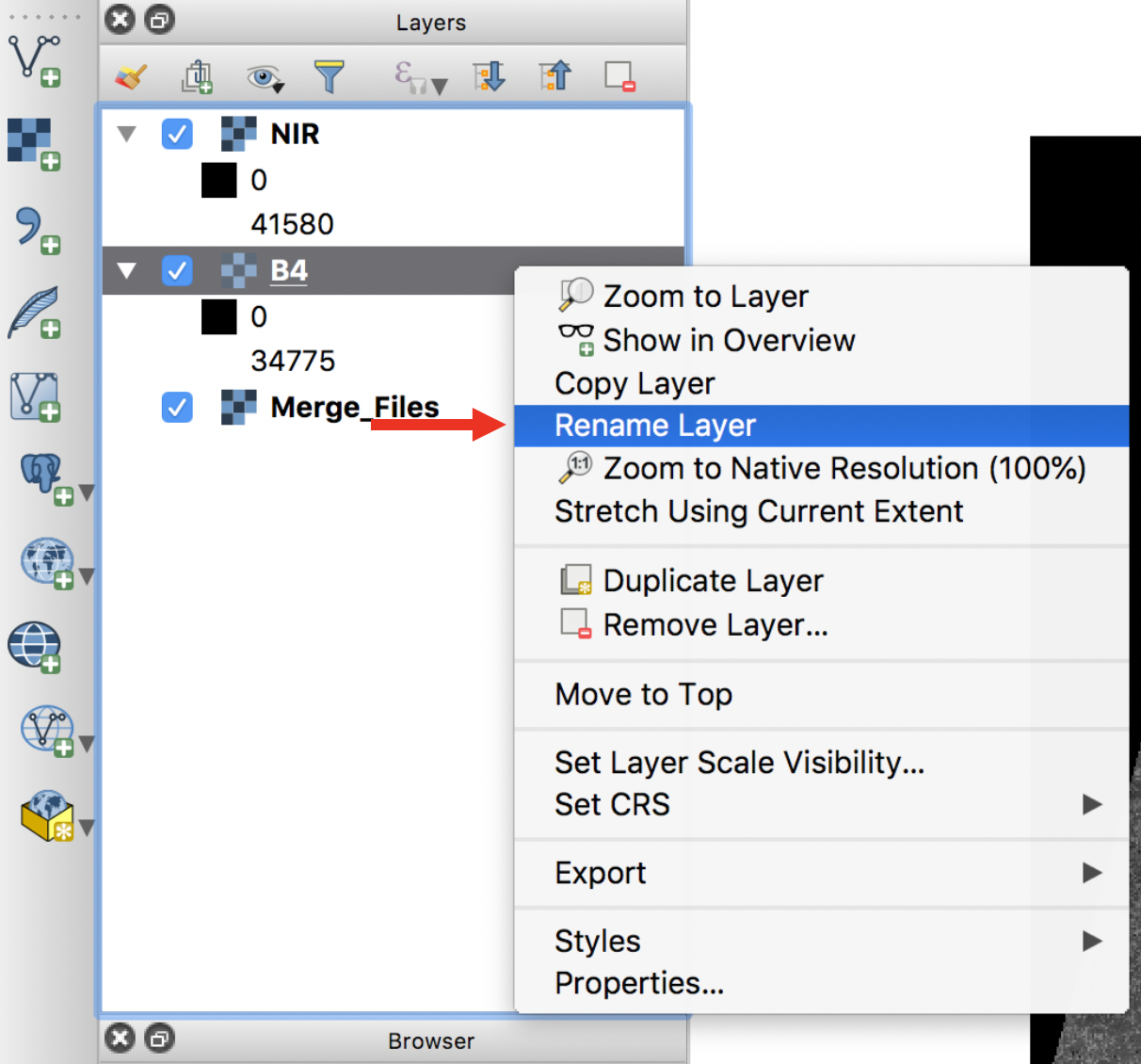
1. The file that has been created can be opened and viewed as a multi-band image in QGIS or other data viewers like R.
2. Practice viewing different RGB combinations of the Landsat bands. In the Table of Contents, right click the composite image and select **properties** or double click the image to pull up the **Layer Properties** dialog. Under Symbology you can assign different bands to display as Red, Green and Blue (Keep Render type as Multiband color). Experiment with different stretches. Also experiment with the Statistics drop-down menu (Min / max values settings).

**Part 3: Create an NDVI image**

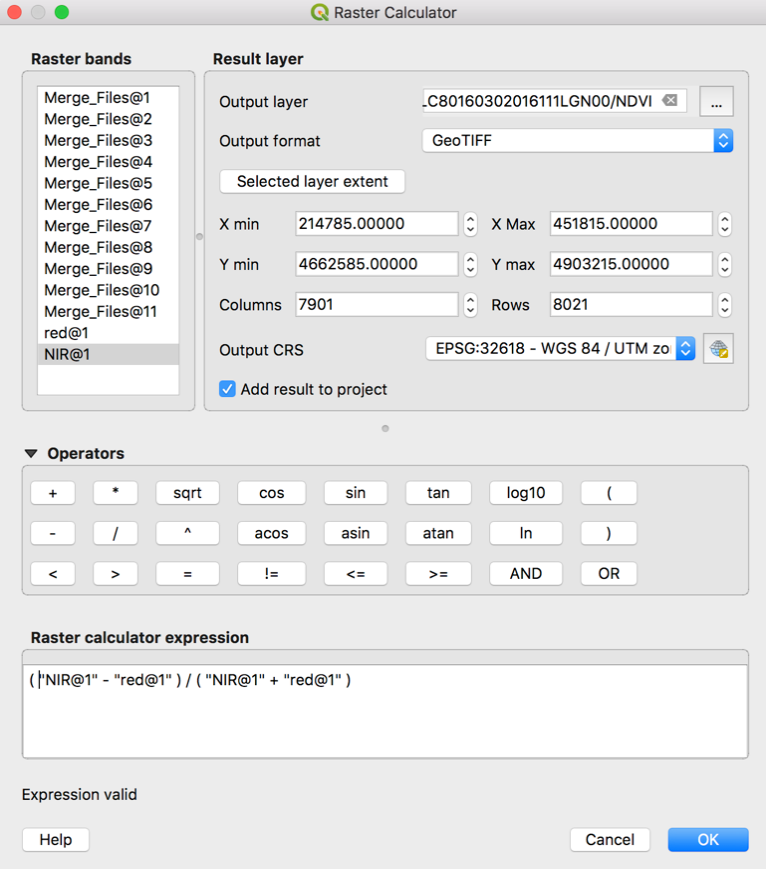
As discussed in the lecture, NDVI is a powerful tool for evaluating vegetation.

Add bands 4 and 5 as layers into your map (say no to creating pyramids). These bands will have \_B4 (band 4) and \_B5 (band 5) at the end of the file name. The individual bands will be displayed as grey scales.

Rename these bands “red” (band 4) and “NIR” (band5) within the map (this will not change the filename on your computer) by right clicking and Rename the band name as displayed under Layers.

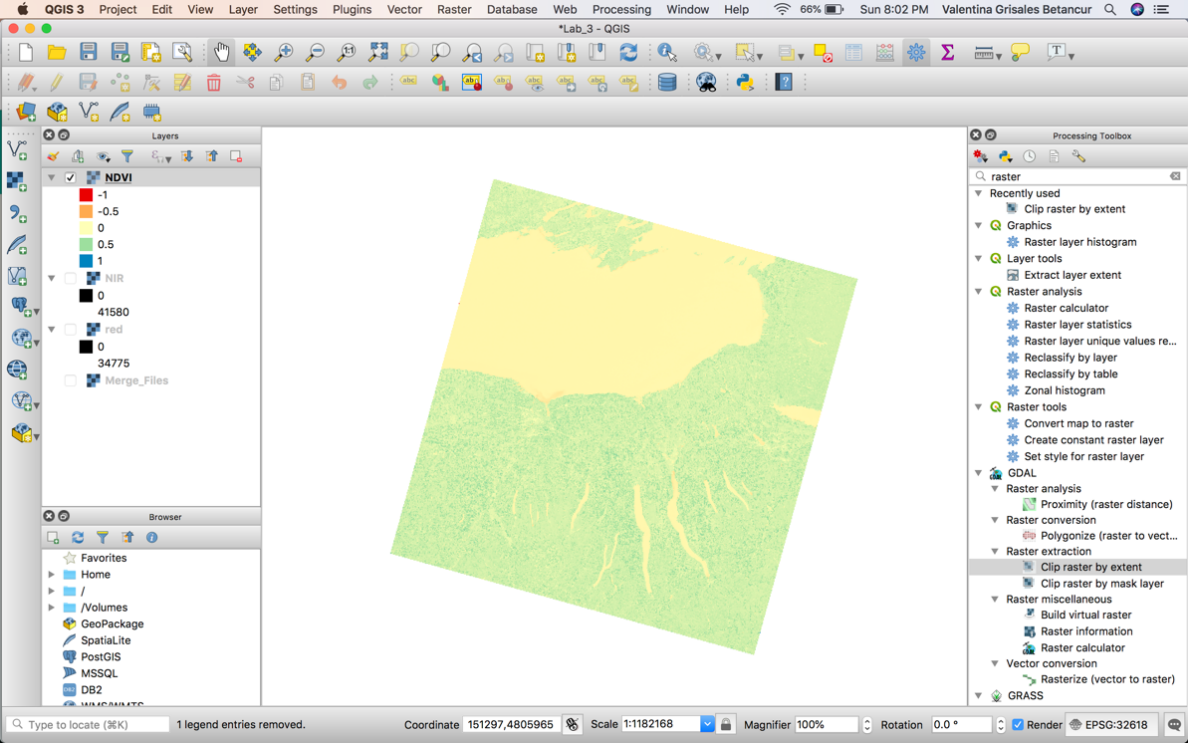


Click on **Raster > Raster calculator**. Notice that this shows you all of the bands in the composite layer, as well as the single bands in Red and NIR. Recall that NDVI is **(NIR – Red)/(NIR + Red)**. Enter this expression into the Raster Calculator ( "NIR@1" - "red@1" ) / ( "NIR@1" + "red@1" ). Save the Output layer as “NDVI.tif”.



Explore the NDVI file, try displaying it with different colors (such as a rainbow gradient) in order to highlight the vegetation response.

**Note: The Landsat scenes, as received from USGS, require additional radiometric and geometric corrections for data specific analysis purposes. This is beyond the scope of this lab and is not easily done using GIS software. However the products created today are suitable for visual interpretation of ground features.**



BONUS ASSIGNMENT – CREATE AN NDVI IN R AND COMPARE IT TO NDVI CREATED IN ARCGIS

[Credit for R code to Ned Horning, CBC-AMNH]

**Part 1:**

If you do not already have it, first you’ll need to download the R software package from [*http://cran.r-project.org/*](http://cran.r-project.org/). If you do not have any experience with R we recommend that you also download the R Manuals “An Introduction to R” and “R Data Import/Export” so that you can explore the basics of the R language. You may also want to download a workspace viewer for R such as R-studio *(http://www.rstudio.com/ide/)*, which allows you to view your code, your data, your file workspace, and graphics/figures at the same time. R-studio also includes some user-friendly tabs that make it much easier to install packages, navigate within your workspace, and import data.

Next, install the necessary packages and libraries in R using the appropriate R script provided in the Session3\_data folder on the course webshare (credit for these scripts goes to Camilo Sanin).

**Part 2:**

install.packages('raster')

library(raster)

## Load in the rasters and turn them into a rasterbrick with 11 bands.

# The bands we downloaded are as follows.

# Band 1 - Coastal aerosol

# Band 2 - Blue

# Band 3 - Green

# Band 4 - Red

# Band 5 - Near Infrared (NIR)

# Band 6 - SWIR1 (short-wave infrared)

# Band 7 - SWIR2

# Band 8 - Panchromatic (detects high-altitude cloud contamination)

# Band 9 - Cirrus (detects high altitude cirrus cloud wavelengths)

# Band 10 - Thermal Infrared (TIRS) 1

# Band 11 - thermal Infrared (TIRS) 2

# Set the working directory to where you saved these rasters

setwd("/pathway/to/your/landsat/rasters")

# Load the rasters as individual bands. You will likely need to change the file names for the rasters you downloaded. B1 = Band 1, etc. Type

# in the object names and look at the attributes of each one. You might notice that some are different resolutions. Luckily, we aren't interested in

# these rasters, so we will leave these out of further analyses for now.

b1<-raster("LC80160302016111LGN00\_B1.TIF")

b2<-raster("LC80160302016111LGN00\_B2.TIF")

b3<-raster("LC80160302016111LGN00\_B3.TIF")

b4<-raster("LC80160302016111LGN00\_B4.TIF")

b5<-raster("LC80160302016111LGN00\_B5.TIF")

b6<-raster("LC80160302016111LGN00\_B6.TIF")

b7<-raster("LC80160302016111LGN00\_B7.TIF")

b8<-raster("LC80160302016111LGN00\_B8.TIF")

b9<-raster("LC80160302016111LGN00\_B9.TIF")

b10<-raster("LC80160302016111LGN00\_B10.TIF")

b11<-raster("LC80160302016111LGN00\_B11.TIF")

# Create a rasterbrick from the list of cropped rasters that all have the same extent.

# This may take some time. To save time you can load fewer rasters, or remove some.

# We only need bands 4 and 5. Use rm(c(b1, b2, b3...)) to remove those rasters from R's memory, then modify the code below to accomodate these changes.

LandSat.stack<-stack(c(b1, b2, b3, b4, b5 ,b6 ,b7))

## Learn about the raster package by gaining basic information about image object

print(LandSat.stack)

#This is the same as

LandSat.stack

#Get class information

class(LandSat.stack)

#Get structure information

str(LandSat.stack)

#Get layer names

names(LandSat.stack)

#Get summary statistics

summary(LandSat.stack)

#Get number of layers in the image

nlayers(LandSat.stack)

#Get extent of image

extent(LandSat.stack)

#Plot an image (this may take some time)

plot(LandSat.stack, 4)

## Using this composite multiband rasterbrick, we want to calculate NDVI.

# Remember: NDVI = (NIR - Red)/ (NIR + Red)

# Bands can be identified using double brackets [[]] (multiband[["LC80160302016111LGN00\_B5"]]) to index the brick. Get in the habit of calling

# rasters by their full name (which could be shortened in the folder they in which they are saved).

# An easy shortcut in RStudio is to hit the tab key within the brackets. This will show you the options of names to choose.

# Remember: in your data will likely have different band names than this example.

NDVI<- (LandSat.stack[["LC80160302016111LGN00\_B5"]] - LandSat.stack[["LC80160302016111LGN00\_B4"]]) / (LandSat.stack[["LC80160302016111LGN00\_B5"]] + LandSat.stack[["LC80160302016111LGN00\_B4"]])

# We could have also done this simply by using the R object names that we gave these files earlier

NDVI\_alt<- (b5 - b4) / (b5 + b4)

# Plot the NDVI layer

plot(NDVI)

# To save the NDVI raster, use the raster function writeRaster(). Check out writeFormats() to see what type of rasters you can save.

# This function will save the raster into your current working directory.

writeRaster(x = NDVI, filename = "NDVI\_R", format='GTiff')

Now, you can also open your new exported ndvi.tif file in QGIS and compare it to your NDVI created in QGIS directly. Are they the same?