Pontus Olofsson, Christopher E. Holden, Eric L. Bullock
Boston Education in Earth Observation Data Analysis/
Department of Earth & Environment, Boston University

1. Introduction

1.1 Overview

The main goal of this workshop is to construct a single-date land cover map by classification of a Landsat image, construct a land cover change map and to estimate areas of land cover change from a sample of reference observations. We will do most of our work on the Windows PCs in room 435 on the fourth floor.

The tools for completing this work will be done in a Virtual Machine (VM) that hosts a suite of open-source tools and that can be installed on Windows, OS X, Solaris and Linux operating systems. The VM is a VirtualBox disk image of the 14.04 LTS release of Ubuntu Mate distribution (https://ubuntu-mate.org/) as of June 10, 2015. This image is stored as an archived Open Virtualization Format (see: http://en.wikipedia.org/wiki/Open_Virtualization_Format for reference). In case shared folders, USB devices, etc. don't mount automatically, you may need to install the "Guest Additions" which is very simple and full instructions are available here: https://www.virtualbox.org/manual/ch04.html. In addition to the default Ubuntu applications, a suite of useful software is pre-installed on the VM, including:

> *Git*
> *GDAL*
> *Orfeo ToolBox*
> *QGIS*
> *Python* (including many scientific Python libraries)
> *R*
> *RStudio*

We assume that you are not familiar with any of these tools and one of the aims of today's assignment is to get an introduction to the software environment. This involves learning how to display and stretch imagery, stack image bands and do some basic image band arithmetic. In this and other lab handouts, menus, buttons, dialog boxes and other components of the software are *italicized*. Folder and file names on the lab computers referred to in the handouts are put in "quotation marks".
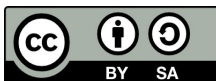
1.2 Install

1. Download a binary of *Oracle VirtualBox* compatible for your operating system from
   https://www.virtualbox.org/wiki/Downloads; follow the instructions to install.
2. Download the latest version of the VM from http://earth.bu.edu/public/ceholden/VM/
   (as of June 24, 2015 the latest version is "Ubuntu_Mate_14.04_20150610.ova")
3. Open *VirtualBox* and click *File > Install Appliance* and browse to the directory containing
   the downloaded ".ova" file. Once installed the ".ova" file can be deleted.
4. Before launching the VM, it is necessary to pay attention to the memory allocation: In
   VirtualBox, highlight the VM in the left-hand pane and click *Settings* (yellow cogwheel) >
   System> Motherboard tab > increase *Base Memory* on the scale line; the amount of
   allocated memory will depend on the host computer but it is recommended to use
   slightly more than half of the memory of the host computer (upper part of the green
   part of the *Base Memory* scale line).
5. Another useful feature is *Shared folders* which are directories on the host computer or
   mounted devices that can be accessed from within the VM (although not technically not
   the same thing, in this and other modules the terms *folder* and *directory* means the same
   thing). You can add shared directories from *Settings > Shared Folders*.
6. In *VirtualBox*, highlight the VM in the left-hand pane and click *Start* (green arrow). This
   will launch the VM; the username and password are both "opengeo-vm" (without
   quotation marks).
7. To view in fullscreen mode click *View > Switch to Fullscreen* in the VM top menu.
8. The interface of the VM can be changed to style of Windows and OS X and various Linux
   distributions by the *Menu* in the upper left corner > *Preferences > MATE Tweak >
   Interface* (the Windows-style interface is referred to as Redmond and the OS X interface
   as Cupertino according to the locations of the headquarters of Microsoft and Apple).
9. Some of the more useful applications are the *Caja* file browser (similar to the file
   browsers in Windows and OS X), the *MATE Terminal* which provides text-based access
   to the operating system, *QGIS* which is the main graphical user interface used in this
   training material, the *LibreOffice* suite which is similar to *Microsoft Office*, and *Atom* or
   Pluma both ASCII text editors. You can add these software to the quick launch panel by
   right clicking the name in the menu > *Add to panel.*
10. Make yourself acquainted to the VM and these software.

1.3 Using the terminal

1. QGIS provides a graphical interface for executing many useful programs and tools but
   there are a few programs, including some very useful GDAL programs, that need
   accessing via the terminal. For these reasons and many others, learning to use the
   terminal is important. For users without Linux and Unix experience this can be a bit
   intimidating at first but it's not hard

2. Open *MATE terminal*: you will see black window saying
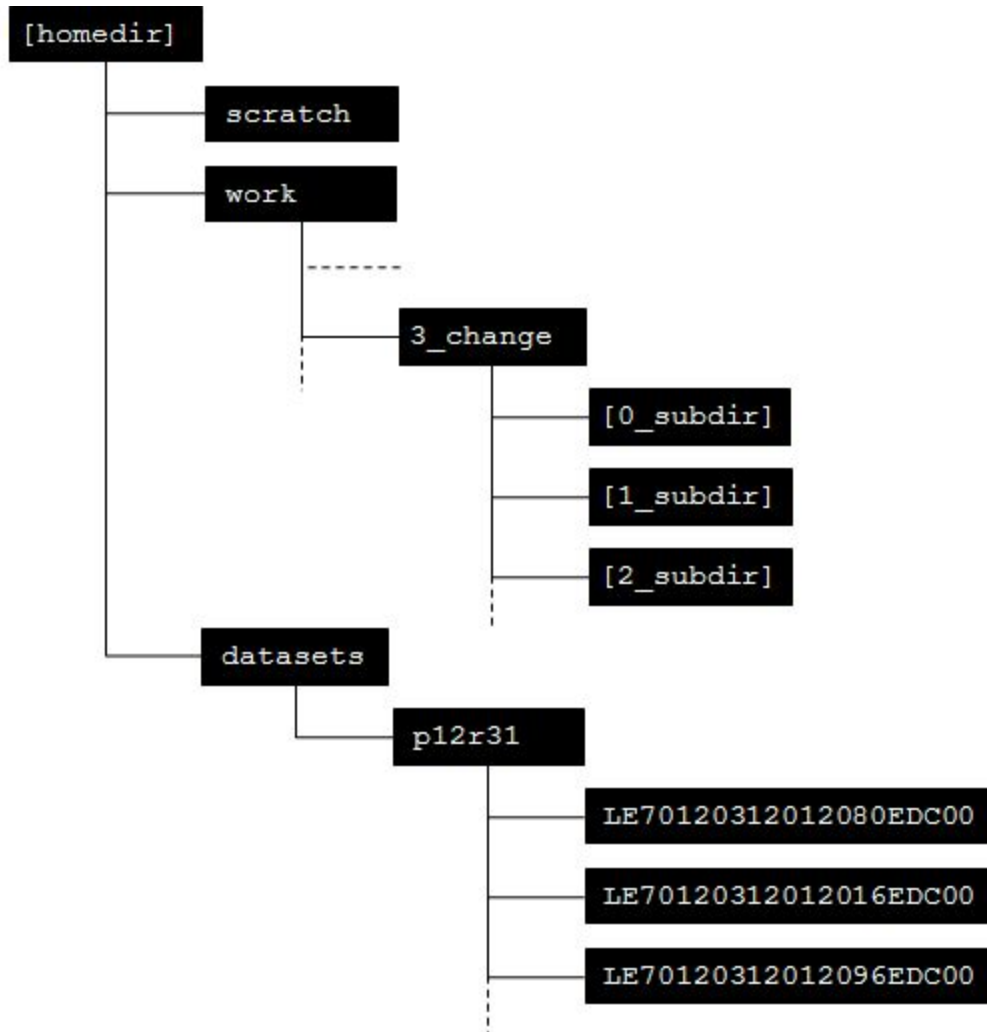   `opengeo-vm@opengeo-vm:~$`
   This means username "opengeo-vm" at the host of the server with the same name; "~"
   means you are currently in your home directory; "$" is the prompt symbol. There are
   many good terminal tutorials online, like this one for example:
   https://www.digitalocean.com/community/tutorials/an-introduction-to-the-linux-terminal

   Spend half an hour to complete this or another online terminal tutorial if needed.

1.4 Data organization

1. Keeping your data organized will greatly facilitate your work. We suggest the following
   structure: in your home directory, which might be a shared folder, a mobile disk or the
   home directory in the virtual machine, create three directories: one for scratch files
   (temporary files, test files, etc.), one for the datasets that you will use (Landsat data for
   example) and one for working on the assignments. These directories will be referred to
   as your scratch, data and work directories. For the data directory, it is recommended to
   create subdirectories for each Landsat path and row, with each image in separate
   subdirectories with the path and row directory; and that each assignment has its own
   work subdirectory ("1_into, 2_classification, 3_change, ..."). Further instructions are
   provided in each module regarding the assignment structure (referred to as "0_subdir,
   1_subdir, …" below). This gives structure as depicted below:

```
[homedir]
    ├── scratch
    ├── work
    │       ┆
    │       └── 3_change
    │                 ├── [0_subdir]
    │                 ├── [1_subdir]
    │                 └── [2_subdir]
    │                      ┆
    └── datasets
              └── p12r31
                      ├── LE70120312012080EDC00
                      ├── LE70120312012016EDC00
                      └── LE70120312012096EDC00
                           ┆
```

2. Create the structure. In this example, my home directory is /home/opengeo-vm/demo; navigate to your home directory and create a directory called "datasets" in the *MATE Terminal*. If you didn't learn how to do it in Subsection 1.3, type:

```
opengeo-vm@opengeo-vm:~/demo$ cd ~
opengeo-vm@opengeo-vm:~/demo$ mkdir datasets
opengeo-vm@opengeo-vm:~/demo$ cd datasets
opengeo-vm@opengeo-vm:~/demo/datasets$
```

3. Create your working directory. If you didn't learn how to do it in Subsection 1.3, type:

```
opengeo-vm@opengeo-vm:~/demo/datasets$ cd ..
opengeo-vm@opengeo-vm:~/demo$ mkdir work
opengeo-vm@opengeo-vm:~/demo$ cd work
opengeo-vm@opengeo-vm:~/demo/work$
```

4. Create a scratch directory. If you now navigate to your home directory and type `ls -l` to list the directory content, you should see the following:

```
opengeo-vm@opengeo-vm:~/demo$ ls -l
drwxrwxr-x 2 opengeo-vm opengeo-vm 4096 Jun 24 11:24 datasets
drwxrwxr-x 2 opengeo-vm opengeo-vm 4096 Jun 24 14:05 scratch
```

```
drwxrwxr-x 7 opengeo-vm opengeo-vm 4096 Jun 24 14:05 work
```

1.5 Extract Landsat data

1. Create a directory in "work" called "1_introduction". In "1_introduction", create six subdirectories: "0_sourcedata", "1_extract", "2_stack" , "3_ndvi", "4_subset", and "5_composite".
2. Copy a Landsat tarball (a compressed Landsat image with extension "tar.gz") to "0_sourcedata".
3. You can extract the data using either the *Caja* file browser or *MATE Terminal*: to use *Caja*, right click the tarball > *Extract to…*; this will open the Extract dialog window > navigate to "1_extract" > click *Create Folder* to create a directory with the name of the Landsat tarball > click *Extract*. It should look like this in Caja:
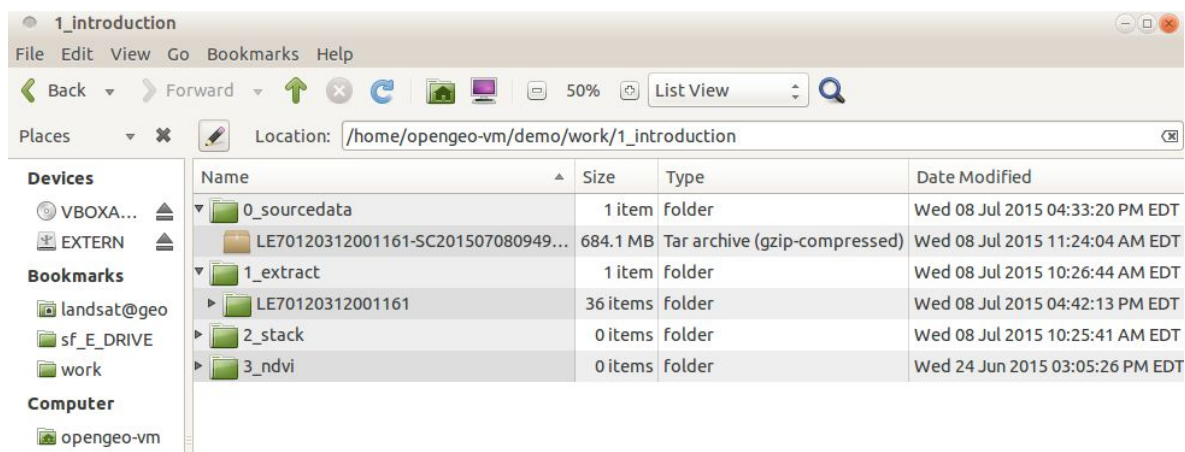
with the contents of the subdirectory in "1_extract" (in this example "LE70120312001161") containing all of individual Landsat bands. In MATE terminal, you will need navigate to "1_extract", create the image directory and type `tar -xvf [tarball]` to extract the tarball (a very useful trick when using the terminal is to type the first letter of directories or files and hit the tab key to autocomplete, this is especially helpful when completing long filenames like that of the tarball):

```
opengeo-vm@opengeo-vm:~$ cd work/1_introduction/1_extract/
opengeo-vm@opengeo-vm:~/work/1_introduction/1_extract$ mkdir
LE70120312005204
opengeo-vm@opengeo-vm:~/work/1_introduction/1_extract$ tar
-xvf
/home/opengeo-vm/work/1_introduction/0_sourcedata/LE7012031200
1161-SC20150708094915.tar.gz
```

1.6 Stack Landsat bands

1. The main interface for analyzing geographical data in this training material is *QGIS*. It is similar to proprietary software like *ArcGIS* but it's free and open source and allows for integration of user-contributed plugins, *GDAL*, *Orfeo Toolbox*, etc
2. To create a layer stack of your images, first open *QGIS*. After the application opens, click *Raster > Miscellaneous > Merge*; this will open the *Merge* dialog > click *Input files* to select Landsat bands 1-5 and 7 in "1_extract".*.*
3. Click Output files and navigate to "2_stack" and provide an appropriate file name like "LE70120312001161_stack" for example.
4. In the *Merge* dialog, leave all options unchecked except for *Layer stack > OK*. This will add the stacked image to the QGIS canvas.
5. Click *Raster > Extraction > Clipper*; this will open the *Clipper* dialog > specify the newly created stack as input and the specify an output file, for example "LE70120312005204_stack_subset". Select *Extent* as *Clipping mode* and draw a rectangle using the mouse on the canvas > click *OK* in the *Clipper* dialog to create the subset.
6. Note: This is also possible via the command line using gdal_merge.py. When doing this, the '-separate' option must be specified to create a layer stack (see http://www.gdal.org/gdal_merge.html for documentation).

1.7 Display image

1. If not already added to the QGIS canvas, open the Landsat image located in "~/work/1_introduction/2_extract" from *Layer > Add Layer > Add Raster Layer* or click to the left of the *Layer* pane.
2. To change the color composite and stretch the image, right click the stacked image name in the layer pane > *Properties > Style*; the window below should appear; set *Render type* (a) to multicolor and the *Red*, *Green* and *Blue* band to bands 5, 4 and 3 (b-d), set *Extent* to *Full* (e), *Accuracy* to *Estimate* (f) and click *Load* (g) > *OK* to display a stretched 5-4-3-false color composite of the Landsat image.
   NOTE: if you're displaying an atmospherically corrected image downloaded from the USGS the no data value is -9999 which you will need to specify in *Properties > Transparency > Additional no data value* before stretching the image*.*
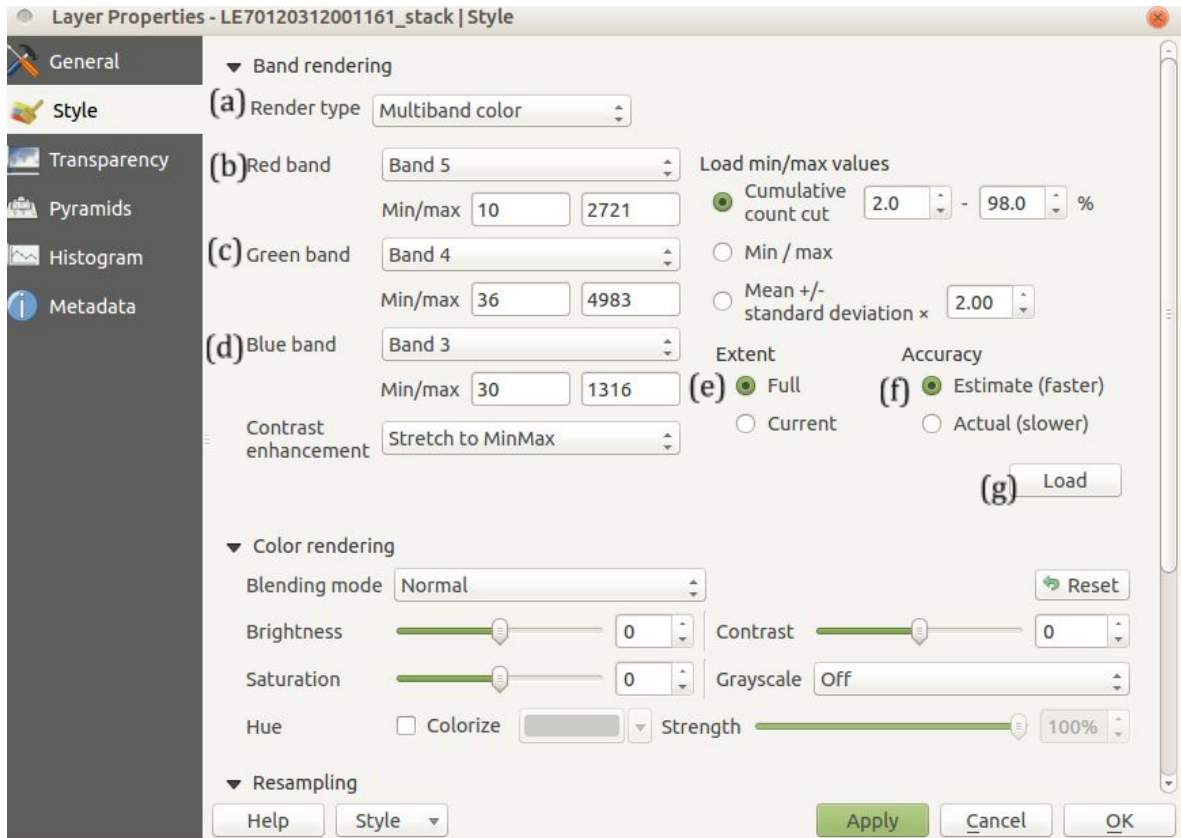
3. Use the different zoom and pan tools to get acquainted to the interface. QGIS has many different tools for navigating an image, but here are a few to get you started:

Button	Name	Purpose
	Zoom in	Zoom in to a selected area
	Zoom out	Zoom out of a selected area
	Zoom full	Zoom to the full extent of the highlighted layer
	Zoom back	Zoom to the previous extent
	Edit	Edit the highlighted vector layer. This can be used to edit attributes or add and delete features.

	Identify Features	View the attributes of the highlighted layer at the specified location. This works for either raster or vector layers.
	Pan Map	Pan around the map without selecting any layers.
	Save	Save the current QGIS project.

4. Now try to use to use the *Raster > Raster Calculator* to calculate NDVI which equals (band 4 – band 3) ÷ (band 4 + band 3) and apply an appropriate color ramp and make sure you save it to "3_ndvi"; it should look like below:
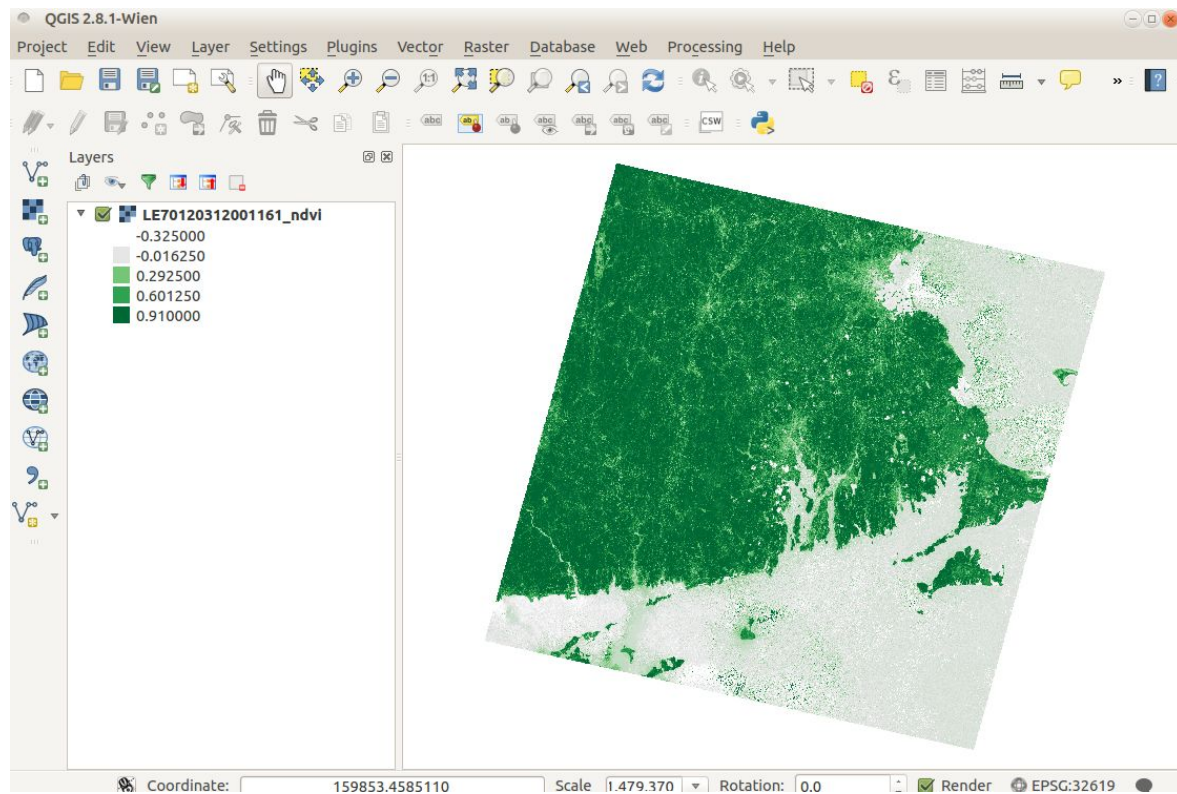
1.8 Create a subset

1. When testing new algorithms or forms of analysis, computationally it makes sense to start with a subset. This makes it faster to repeat the analysis with different parameters or options.
2. With your raster image open in QGIS, go to *Raster > Extraction > Clipper.*

3. For the *Input file* choose your Landsat image.
4. Save the *Output file* to "1_Introduction/4_subset".
5. For *Clipping mode* choose "Extent".
6. Now draw on the raster layer the extent you want to subset. If you click and drag with the mouse it will automatically draw a bounding box. Size does not particularly matter, but try to make it big enough to capture multiple land covers. Alternatively, if you have coordinates you wish to use you can enter them in the boxes provided. If not, they will automatically be filled in by the bounding box you draw.

1.9 Image compositing

1. You might consider making a composite from a group of images. This composite will contain the "best" possible pixels from all the images in a time period. Often, it is hard to find a single image cloud-free image to do your analysis. In these occasions, a composite can be preferable to any single real image. Numerous compositing algorithms exist, but they all rely upon the idea of using an indicator to decide which image to take the pixel values from. This indicator includes: Maximum NDVI, median value for each band, and maximum NIR / blue.
2. To make a composite image, you need a number of images in "5_composite". These images must be extracted and stacked to matching extents.
3. First, the selected images must be extracted into individual folders.
   a. Use LandsatLook (http://landsatlook.usgs.gov/viewer.html) to locate three images in your Path Row that are relatively cloud free and close to each other in time. Write down the image IDs.
      i. Zoom in to your location.
      ii. Under *Select the Landsat Archive* choose the time period, sensors, and maximum cloud cover you wish to search for.
      iii. Click *Show Images*
      iv. Use *Image Display* to sort through images. Look for 3 images that are near in time and relatively cloud free. Write down the dates.
   b. Find the tarballs of the images you choose and copy them to the "1_extract" folder.
   c. Once in the "1_extract" folder, make individual folders for the images to extract. Folder names should correspond to the image (ex: 'LE70120312001161' if the tarball is named 'LE70120312001161-SC20150708094915.tar.gz').
   d. To extract the files into the directories, specify the '-C' option in the tar command:
      ```
      tar -xvf [tarball] -C [directory]
      ```
   e. An example of, the command would be:
      ```
      tar -xvf LE70120312001161-SC20150708094915.tar.gz -C
      LE70120312001161
      ```
4. To conserve on file space, instead of copying the folders to "5_composite" you can create a **Symbolic Link** to the folder..
   a. The command in Linux to make a symbolic link is:
      ```
      ln -s [directory] [target link location]
      ```

b. Note, you need to be in the [target_link_location] while creating the link for this to work, so for this exercise you must be in the '5_composite' folder.

c. To link the folder in the previous example to a new folder in "5_composite", the following command will be used while in the "5_composite" folder:

```
ln -s ../1 extract/LE70120312001161 ./
```

5. The images must then be stacked to matching extents.

a. Stack each image using the steps in **1.6**.

b. To ensure matching extents, clip each raster using the same vector file. To do so, please **Refer to subsection 2.1 in Supplementary Tutorial S2** *Methods Classification*.

6. To perform the compositing, we have written a Python script.

a. Copy "image_composites.py" from your "scripts" folder to "5_composite".

b. To look at all the options in the script, type:

```
python image composites.py -h
```

c. The basic syntax is:

```
python image composites.py -algo [algorithm] -o
[output name] [image1] [image2] [image3] [etc]
```

d. Instead of writing every image that you wish to composite, an alternative is to give it a folder and file pattern, using asterisks to in place of the characters that differ between the files.

e. To make a composite of 'image1.tif' 'image2.tif' and 'image3.tif' (within the folders image1, image2 and image3) with a 'Maximum NDVI' algorithm, use the following command:

```
python image composites.py -algo maxNDVI -o
image composite.tif i*/i*tif
```

f. The following algorithms are implemented in the script: maxNDVI, maxNIR, ZheZhu (maximum NIR / Blue), medianNDVI, minBlue.

i. Note: The script also has the option to write your own algorithm if desired (see help documentation).

g. The output file will be an *n*-band image with *n* equal to the amount of bands in the input images. Each pixel will correspond to the original pixel values that meet the criteria of the composite algorithm.