# GEOG60941: Environmental Remote Sensing
# Practical 5: Landsat time series analysis using LandTrendr

**Aim**: The aim of this practical is to explore LandTrendr (LT) in Google Earth Engine (GEE) for assessing time series of Landsat data. At the completion of the practical you will be able to:

- Understand what LandTrendr is, how it works and what it can do
- Understand how to run LandTrendr through the GEE code editor to create spatio-temporal maps of wildfire disturbance.

**Acknowledgements:** ARSET NASA

Time series analysis is an important technique for analysing changes in the landscape over time. For retrospective analysis of disturbance events, ***Landsat data*** can be used to establish medium spatial resolution patterns of disturbance extent, their duration and magnitude across large regions. There are many types of time series algorithms that can be applied to Landsat data to analyse disturbance. In this practical we will use the **LandTrendr** algorithm, specifically to target disturbance from a large wildfire event. LandTrendr allows users to generate maps of **Year of Detection, Magnitude, Duration** and **Rate of Change** within Google Earth Engine.

**Core tasks for this session:**

1. Map change over time using the LandTrendr Change Mapper GUI interface
2. Map change over time using the LandTrendr Change Mapper via the GEE code editor
3. Export the results of the LandTrendr algorithm from GEE and import them into a GIS ready for further analysis.

**Files required from BB:**

1. KingFireBuffer.zip – shapefile (and associated auxiliary files) of the King Fire boundary. These data can be downloaded from the BB site.

# LandTrendr in Google Earth Engine

---

We will explore LandTrendr (LT) in Google Earth Engine (GEE) for assessing time series of Landsat imagery. LT-GEE is a set of algorithms that can be used for change detection in a time series of images. The benefit of the LT-GEE algorithm is that you can process many Landsat images "on the fly" with the computing power of Google, avoiding the need to download and process large amounts of imagery on your own PC.

For this practical we will use both the LandTrendr *Graphical User Interface* (GUI or UI) (i.e. a web app built on top of GEE) which is primarily designed for those who are not familiar with GEE and coding AND we will run the code directly within the GEE code editor, which allows more flexibility and functionality.

The practical will focus on detecting *rapid changes* caused by a large wildfire, but the algorithm can also be tailored towards identifying slower, more gradual disturbances (or recovery from a disturbance) too.

Before you begin, it's important to review the LandTrendr information here: https://emapr.github.io/LT-GEE/landtrendr.html. Also, for details about each LT-GEE UI feature and the output that we produce, refer back to this site as we go through the exercises

There are three UI applications within LT-GEE:
(1) Pixel Time Series Plotter,
(2) Change Mapper, and
(3) RGB Mapper.

For this exercise, we will focus on the **Change Mapper** module both within the user friendly GUI and within the code editor itself. However, feel free to take a look at the other two modules as they may be useful for an initial exploratory analysis of your chosen location.

# Examining change induced by wildfire

We will focus this practical on the King Fire that occurred near Yosemite National Park in **2014**. This fire burned 39,545 hectares of land (Fig 1).

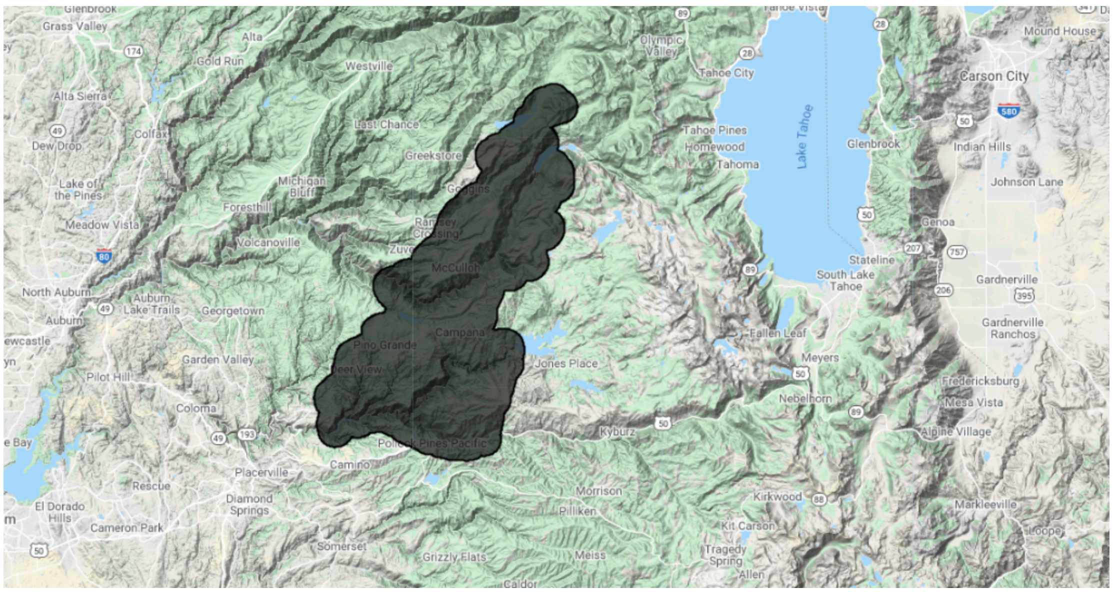



Fig 1: Outline of the King fire boundary

## Task 1: Mapping change over time with “Change Mapper” – Using the GUI

Our first task is to use the GUI to map change over time specifically to identify the Yosemite National Park fire event that took place in 2014. The GUI is useful for learning the key concepts of LandTrendr, but the code editor should ultimately be used to do the final processing.

We will use a time series of annual NDVI values (one image per year) to identify the fire disturbance event. Since in this case, we know the date of the fire then we can set the range of our time series as **2012** and **2016** to capture this event; using trajectory analysis (i.e. LandTrendr)

1. Open Google Chrome and go to the User Interface LT-GEE website: https://emapr.github.io/LT-GEE/ui-applications.html. Here you will find a whole host of information relating to the LandTrendr user interface tool. As previously noted, for this part of the practical we will focus on using the **Change Mapper** tool. Scroll down to section **8.2 UI LandTrendr Change Mapper** and read the information about the tool i.e. what it is and what it does. The output from this tool will include a change map and additional attributes such as year of change detection, magnitude of change and duration of change event.

2. Scroll down to the **8.2 UI LandTrendr Change Mapper**, click on **GEE App link** (directly under the subheading title) and this will take you directly to the *User Interface* page (Fig 2)

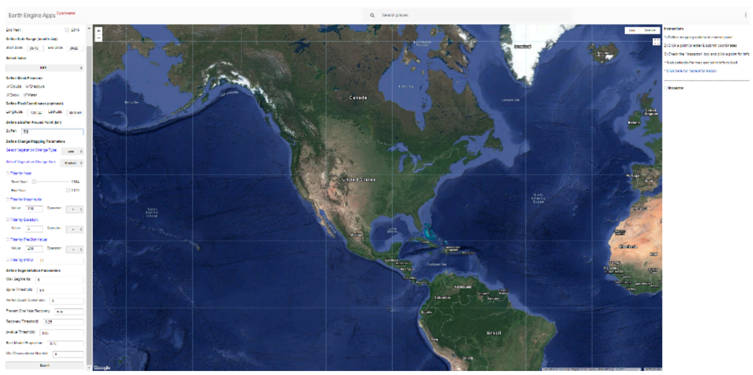


Fig 2: Example of the Change Mapper GUI

3. Review the features on the left-hand side panel to make sure that you understand the optional settings available. If you have questions, you can always refer back to the UI main page.

4. To generate change data for the King Fire, select the following options:
    a. Leave the **Define Year Range** over which to build a Landsat time series to identify change as it is - it's best to set this close to the maximum range as you can filter changes by year in a different setting below.
    b. We'll also leave the **Define Date Range** box as the default option as we're working in the northern hemisphere and so this range should be fine to avoid too much cloud cover or seasonal change.

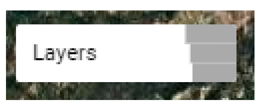
c. Under **Select Index**, select **NDVI** as the chosen index for this exercise - you can expand the box to see some other predefined index and individual spectral band options.
d. Leave the **masking elements** as they are as we want to try and mask out clouds, shadows, snow and water.
e. Under the **Define Pixel Coordinates (optional)** section, enter the coordinates where we want to search for our fire event **-120.5479** next to **Longitude** and **38.9089** next to **Latitude.**
**f.** Under **Define a Buffer area around a point,** change that value to **2** (km**)** because I know that this particular fire event occurred within 2 km of our defined location. Landtrendr is quite a computationally intensive algorithm so it's wise to try and narrow down your search area as much as possible, based on *a priori* information. *If your computer takes an excessive amount of time to load the data (it can vary depending on the internet connection), then for the purposes of this exercise, you can reduce this value further.*
g. In the **Define Change Mapping Parameters** box:
    i. Leave **Select Vegetation Type** as **Loss** - we want to identify changes which relate to vegetation loss. You can change this to Gain if you are interested in vegetation gain or growth.
    ii. Leave **Select Vegetation Change Sort** as **Greatest** - we want to identify pixels with the disturbance event associated with the largest change (i.e. greatest losses), but take a look at the other options. *Note: You can only identify one disturbance event within the time series for any given execution of the algorithm.*
    iii. Tick **Filter by Year** box and use the sliders in the left-hand panel to change the **Start Year** and **End Year** sliders to filter the imagery between **2012-2016**. This will allow the algorithm to use imagery prior to and after our disturbance event to fit the disturbance model.
    iv. Tick the **Filter by magnitude** box and change the value to **300** and leave the **Operator** as **>.** This will tell the algorithm to only identify pixels as being "changed" when the NDVI value changes by more than 300 - remember that the index values are scaled by 1000 so this actually means a change in the NDVI value of more than 0.3.

v. Tick the **Filter by Pre-Dist Value** box and change the value to 400 to retrieve only those pixels that had a pre-disturbance event NDVI value of 400 (i.e. 0.4) to try to ensure that the pixels were initially vegetated.

vi. Tick the **Filter by MMU** (*minimum mapping unit*) box and leave the value as 11. This will prevent the algorithm from returning isolated change pixels, which are likely to be erroneous for this type of disturbance event. A value of 11 means that the minimum "patch" size (or mapping unit) of change must contain at least 11 connected pixels. This should be changed depending on what type/size of event you are trying to detect.

h. Leave the **Define Segmentation Parameters** as default for now, but it's important to read the LandTrendr literature/documentation provided so that you know what they do. You may need to tweak these if your model does not fit your data very well.

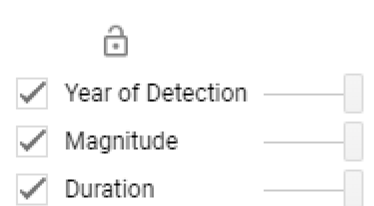
2. Scroll down to the bottom of the left-hand side panel and click **Submit**.

You will be taken to the location of the King Fire and change images will be generated (Fig 3). However, you need to wait for the page to finish loading before you can see the images . This may take a few minutes or longer depending on your connection speed. There are multiple layers being processed at once e.g 

. Notice the Layers box on the top of the map. The grey bars will turn all white when the processing is finished 

. *Try to avoid zooming in or out of the image as this will cause GEE to start recalculating the change layers!*

If it's taking some time you can probably move onto point 5 below whilst the image is still loading as the data are probably already processed, it's the visualisation that takes time. When it's finished you should see three layers appear in the **Layers** box



The **Year of Detection** image will be at the top and visible in your display (as in the figure below).

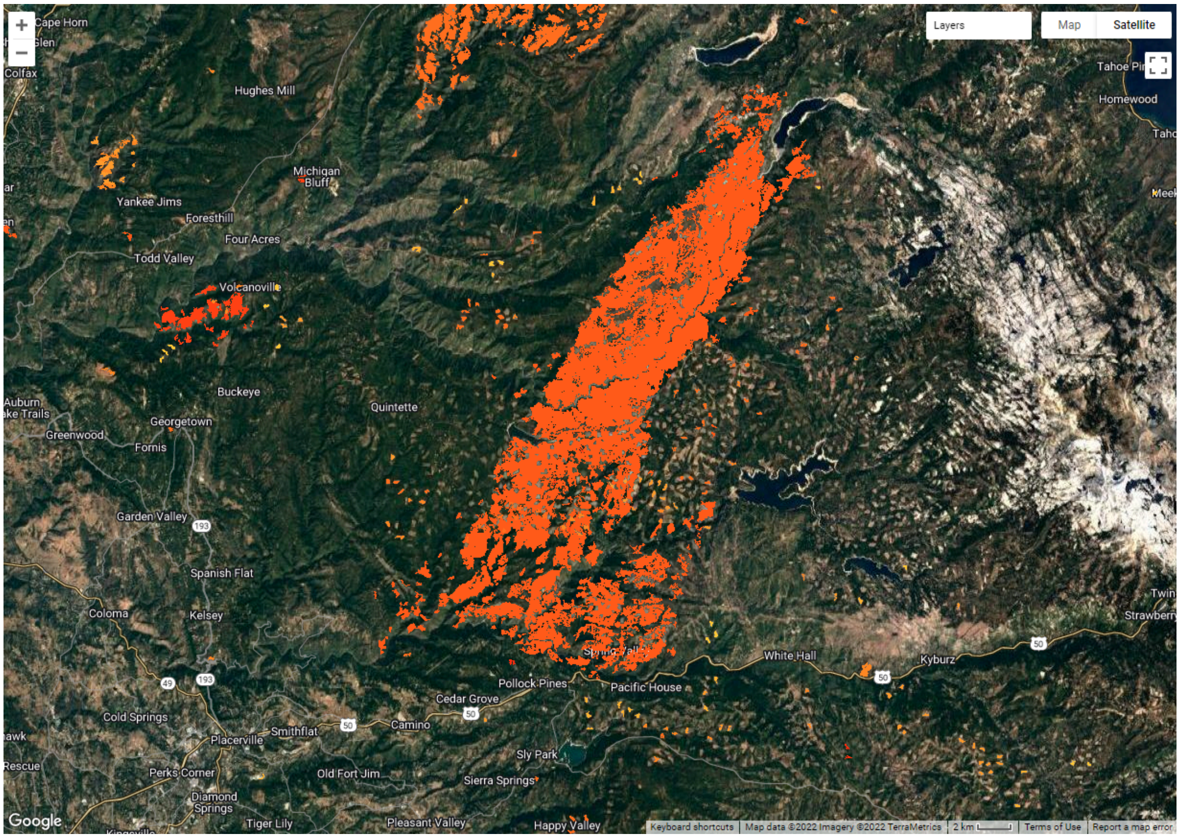



Fig 3: Year of Detection image for the King Fire from LT-GEE

5. Unfortunately, as we already know, GEE doesn't produce any map legends by default, so to see when the disturbances occurred in each pixel (i.e. what year each of the colours represents), click the **Inspector** box on the right and click somewhere in the biggest orange area. This large disturbance is the King Fire.

On the right you will see a plot appear (it may take a little time to load) along with information about the disturbance detected for that pixel, including **Year, Magnitude**, **Duration, Pre-value (index value * 1000 prior to the event) and Rate of change (mag/dur).** Note that the year refers to the first year that a "break" in the NDVI was detected, which was 2015, so the fire occurred 1 year prior (duration = 1) in 2014.

☑ **Inspector**

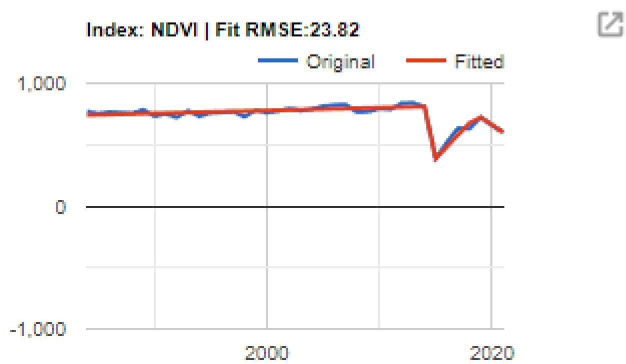
Year: 2015

Magnitude: 424

Duration: 1

Pre-value: 810

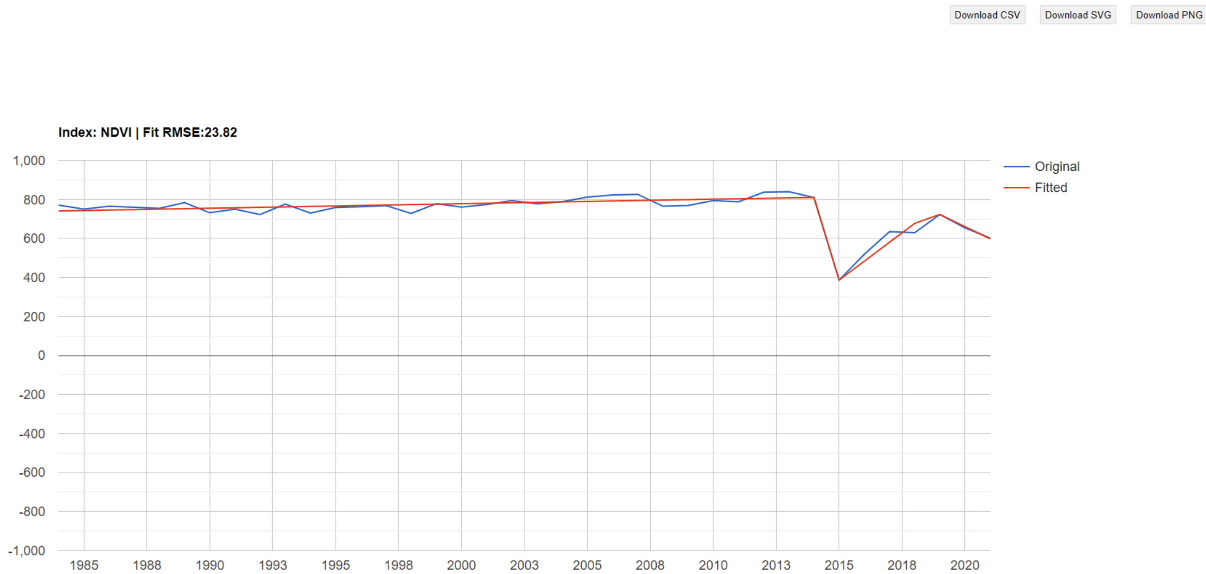
Rate: 424




6. As usual, you can click on the small box with the arrow in the upper right to show the graph in a separate window. The blue line is the actual NDVI data and the red line is the “model” that the LandTrendr algorithm has fitted to the time series. The information that LandTrendr outputs is derived from the “fitted model” so it’s important that the parameter options (that we chose earlier) result in a good model fit. If they don’t then they should be iteratively adjusted (including the segmentation parameters!). You should check multiple pixels to check the fit over space. A good explanation of the fitting parameters can be found [at this link]

7. OK, let’s take a look at this plot in a bit more detail. The NDVI values are plotted on the Y-axis, where a higher value indicates healthy vegetation and a lower value indicates bare ground or recently burned area on a scale of 1 to -1, and the year is plotted along the x axis. Hover your mouse over the plot to show the year and the NDVI pixel value for different points in the time series. *Note: Remember that the displayed values should be divided by 1,000 for*

*ratio and normalised difference spectral indices (all the decimal-based data are multiplied by 1,000 so that the data type can be converted to signed 16-bit and retain some precision).*




You can see that this pixel experienced a high magnitude disturbance event with a negative (loss) change (i.e. delta) in the NDVI index value resulting in a value of ~0.42 (i.e. 424/10000) immediately after the disturbance occurred. However, the disturbance only lasted a short amount of time (duration 1 year) after which the vegetation started to recover. All values are related to the *fitted time series* model, which the algorithm uses! Since high NDVI is indicative of healthy veg and a low NDVI can be indicative of burnt vegetation and this was a sudden and short event, we can suggest that this disturbance might be caused by a fire - in fact we already know that it was and we can clearly see the fire perimeter in the change image.

8. Close the chart and go back to the **Change Mapper.** Hopefully the layers will be loaded by now if they weren't already! You will see that **3 layers** were produced: **Year of Detection, Magnitude, and Duration**

You can use the slider bar beside each layer to adjust their transparency ***but don't turn them off as GEE may start reloading the data when you turn it back on again***! In this image, the legend (although unhelpfully it's not displayed!) for the **Year of Detection image**, ranges from yellows, which indicate earlier disturbances to oranges and reds which indicate later disturbances. This is a highly fire-prone region, so the large patches of different colours primarily represent wildfires occurring between your filtered date range.

9. Make the **Year of Detection** layer totally transparent by sliding the slider to the left. Now you can view the **Magnitude** layer (Fig 4).

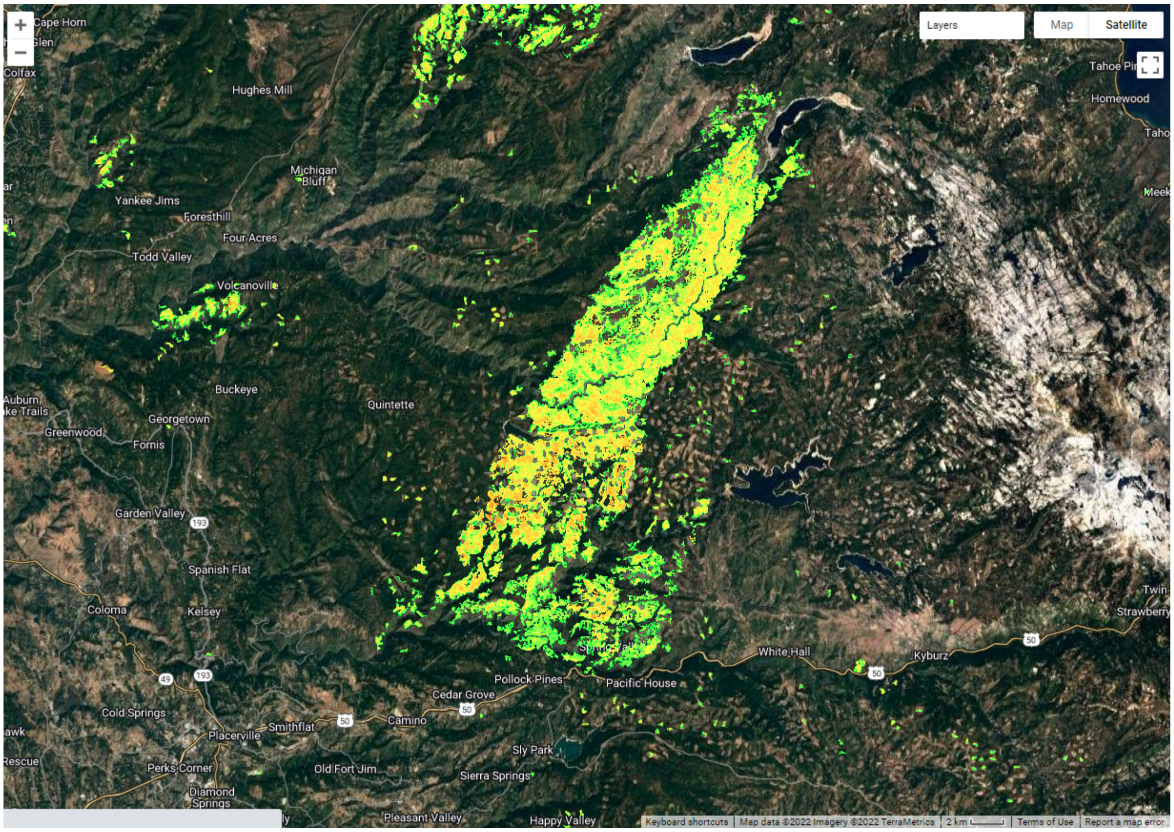



Fig 4. King Fire magnitude of change layer from LT-GEE

The highest intensity burn regions are in red and decreasing intensity in yellow and then green. The magnitude ranges from ~0-1,000 (i.e. 0-1). The magnitude of the disturbance is defined as the distance in spectral values (in this case NDVI values) between the beginning of the detected disturbance and the lowest NDVI value reached after that event.

10. Now make the **Magnitude** layer transparent using the slider, so that you can just see the **Duration** layer.

11. The red and orange colours represent shorter duration disturbances, such as wildfire, while the yellow, green and blue represent longer duration disturbances (there are hardly any in this scene!).

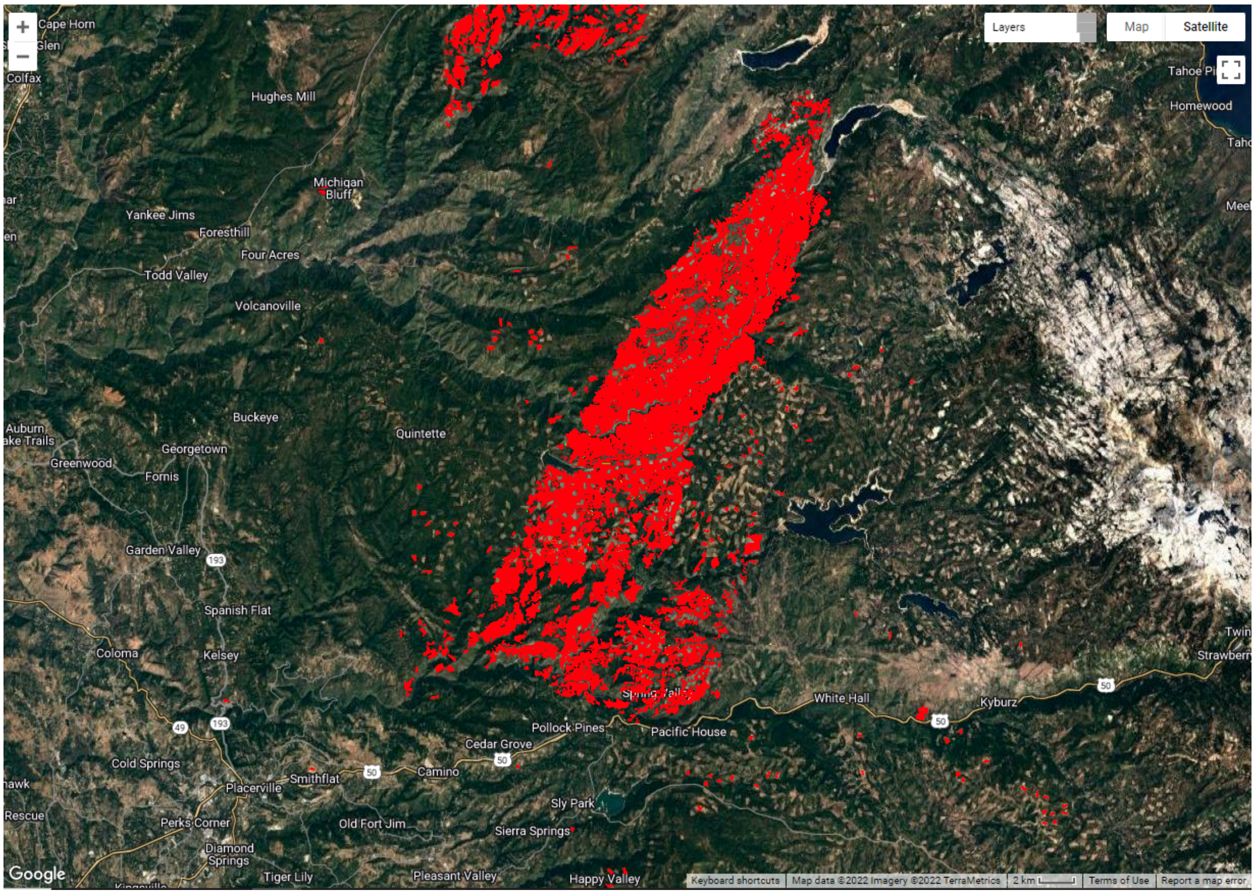



Fig 4. King Fire duration of change layer from LT-GEE

Feel free to explore the GUI further, by perhaps adjusting the parameters to look at vegetation gain rather than loss - how quickly has the vegetation recovered?

The purpose of looking at the GUI is to simply understand the inputs and some of the outputs of the LandTrendr algorithm. The GUI has a number of limitations - particularly the fact that you cannot export any data from it!

If you are planning to use LandTrendr for your assessment then you will need to use the code editor to adjust the underlying LandTrendr code. Consequently, make sure you are happy with what all the parameters/options mean before you move on to task 2.

# Task 2: Mapping change over time with “Change Mapper” – Using the Code Editor

As previously mentioned, whilst the GUI is extremely useful for generating a quick overview of the change and looking at how the model fit varies when you adjust the parameters, one current limitation is the inability to output change *images*. The only way to do this at the moment is to actually enter the code, which underlies the GUI app, into the GEE code editor and output it from there. The code editor also allows you customise the algorithm a lot more to suit your particular application. Seeing as we are already familiar with the code editor, we can do just that.

The LT-GEE documentation provides us with a number of “example scripts” so we don’t have to code it from scratch, rather what we need to do is just edit it, but prior to this **we need to understand what it is doing**. Familiarising yourself with the GUI (as in Task1) will really help you.

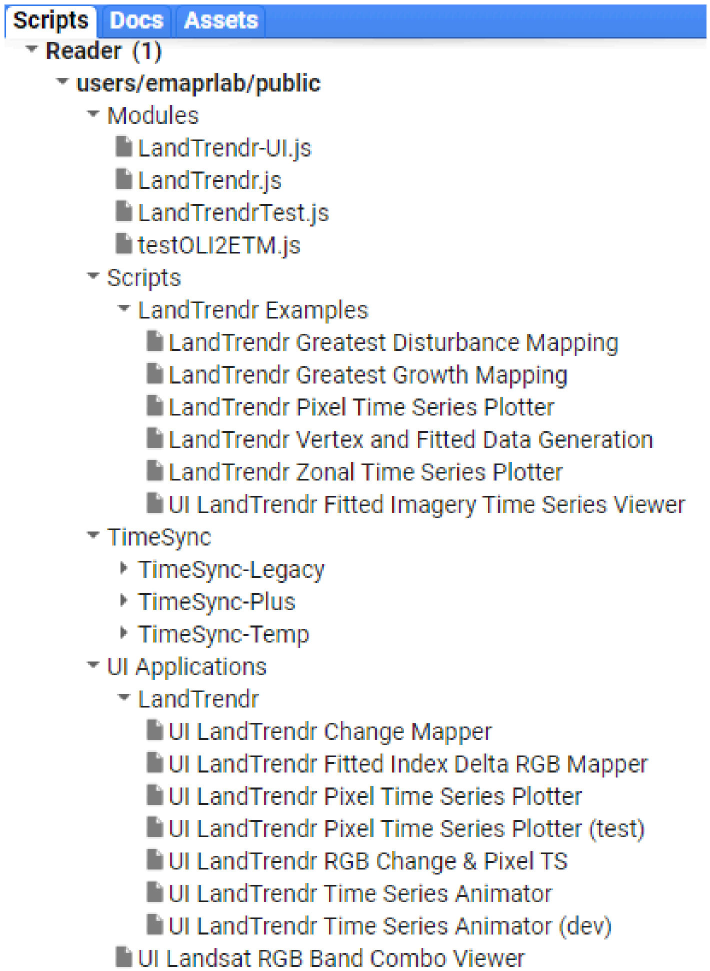
The following web page provides extensive and very useful explanations of how to run, select and adjust the LandTrendr parameters for change detection retrieval within the GEE code editor https://emapr.github.io/LT-GEE/introduction.html. If you are thinking of using this tool for your assessment or dissertation then it is a must read. Essentially there are 6 steps, many of which you will see closely align with the information that we just entered in the GUI:

1. Define starting and ending years of the times series
2. Define an area to apply the LandTrendr algorithm to
3. Define the LandTrendr algorithm run parameters in a GEE dictionary
4. Build an image collection that includes only one image per year, subset to a single band or spectral index (you can actually include multiple different bands/indices - the first will be used to identify the breaks (i.e. segment) and the others will be fit to those segments (i.e. the vertices))
5. Append the image collection to the LandTrendr run parameter GEE dictionary

6. Run the LandTrendr algorithm

In this example we'll repeat (almost!) the King fire analysis that we just did in task 1, so that you can get a general overview of how it works. Essentially, we will hard code most of the parameters that we manually entered in the GUI. The main difference is that we will use a vector file of the extent of the King fire to situate our analyses, as opposed to a single point coordinate. Additionally, we'll export the *change images* and their attributes as GeoTiff files to your Google Drive account so that they can be tidied up and/or analysed further in a GIS.

1. The LandTrendr team has helpfully created a JavaScript module that serves as an API which includes some pre-coded LandTrendr functions into javascript (LT-GEE API), which can then be called and run in GEE (i.e. without the need to create them ourselves!). In order to access them, **you must first visit this webpage** ***prior*** to creating the code. https://code.earthengine.google.com/?accept_repo=users/emaprlab/public
   It will add the ***users/emaprlab/public*** repository to your GEE account. You should see the module added to your ***Scripts tab*** in GEE. You should see that there are **Modules** (with the extension .js) but also some example scripts (which are very useful!). You may want to have a play around with these at some point but for now we'll go ahead and create our own script, which is based on the Greatest Disturbance Mapping example!




*(if this does not work for you then you can access the link from here [https://emapr.github.io/LT-GEE/example-scripts.html](https://emapr.github.io/LT-GEE/example-scripts.html) …..”To add, the API, click this link”)*

2. Now start a new GEE script and let’s load the **King Fire shape file**. This file depicts the extent of the King Fire, including a 2 km buffer outside of the fire’s perimeter. If you haven’t already then download the KingFireBuffer.zip file from the BB site. You do not need to unzip it as GEE can ingest the zipped file as is.

3. Now add the shape file as an **Asset** to GEE. Go to the **Assets tab** in GEE, select **New/Shape files,** drag and drop the KingFireBuffer.zip, select a save location for the data and call it **KingFireBuffer.** Click Upload. You may need to click the refresh button  to view the KingFireBuffer listed within your Assets tab.

4. Now import it into GEE, calling it **KingFire** and display it. There are a couple of ways to do this but the easiest is to hover over the file in the Assets window and click on the horizontal arrow to **Import into Script**. The file will appear in the Imports window at the top of the code editor (it will be called Table by default) but rename it as **KingFire** (remember GEE is case sensitive so check!) and display it on the screen.

```
//##########################################################################
// Displaying OUR KING FIRE PERIMETER VECTOR
//##########################################################################

Map.addLayer(KingFire, {palette: '000000'}, 'KingFire');
Map.centerObject(KingFire, 10);
```

5. Now we need to enter the input parameters for the LandTrendr Change Mapper. We will use the same parameters as we did for the GUI interface example. I have completed some for you, you need to complete some on your own i.e. **<your turn>.** First, we'll start with defining the data collection parameters

```
//##########################################################################
// START INPUTS TO LANDTRENDR
//##########################################################################

// Define Collection Parameters
var startYear = 1984; //Define Start Year
var endYear = <your turn>; //Define End Year
var startDay = '06-10'; //Define Date Range Start Date (month-day)
var endDay = <your turn>; //Define Date Range End Date (month-day)
var aoi = KingFire; //Define the location
var index = 'NDVI'; //Define spectral index
var maskThese = ['cloud', 'shadow', 'snow', 'water']; //Define Mask
Elements
```

6. Now we will define the mapping parameters (*see section 9.1.9 of [https://emapr.github.io/LT-GEE/api.html#functions](https://emapr.github.io/LT-GEE/api.html#functions) for more details on these parameter options*)

```
// Define Change Mapping Parameters. Note that checked:false means that you
do not want these parameters to be applied
var changeParams = {
  delta:  'loss',
  sort:   'greatest',
  year:   {checked:true, start:2012, end:2016},
  mag:    {checked:true,  value:300,  operator: '>', dsnr:false},
  dur:    {checked:false,  value:4,    operator: '<'},
  preval: {checked:true,  value:400,  operator: '>'},
  mmu:    {checked:true,  value:11},
};

// Define Landtrendr Segmentation Parameters
var runParams = {
  maxSegments:            6,
  spikeThreshold:         0.9,
  vertexCountOvershoot:   3,
  preventOneYearRecovery: true,
  recoveryThreshold:      0.25,
  pvalThreshold:          0.05,
  bestModelProportion:    0.75,
  minObservationsNeeded:  6
};


//##########################################################################
// END INPUTS
//##########################################################################
```

7. Next we're going to load and run the **LandTrendr.js module** that we loaded to our **script library** at the start. This module draws information from the parameters that we just defined in our script in order to run the change detection

```
// load the LandTrendr.js module
var ltgee = require('users/emaprlab/public:Modules/LandTrendr.js');


// add index to changeParams object
changeParams.index = index;
```

```
// run landtrendr, it automatically creates a Landsat image stack of the //
aoi based on the parameters defined previously
var lt = ltgee.runLT(startYear, endYear, startDay, endDay, aoi, index, [],
runParams, maskThese);

// get the change map layers
var changeImg = ltgee.getChangeMap(lt, changeParams);

// clip the image
var changeImg = changeImg.clip(aoi);
```

8. Now we will quickly plot a couple of histograms so that we can get a feel for the distribution of some of the values for some of the key outputs e.g. magnitude and duration of change images. We'll calculate the histogram stats on a pixel size of 500 m instead of 30 m (which Landsat data actually are) as it's a big image and we just want to take a quick look at the histograms to get a feel for the data - remember that these histograms are just quick looks!

```
//***************************************************************
//                          Visualising the outputs
//*****************************************************************

//Plot some histograms to quickly look at the approx. number of pixels in
different categories/bins
var magOutputHist
=ui.Chart.image.histogram(changeImg.select('mag').multiply(0.001), aoi,
500, 10, 0.1);
var durOutputHist=ui.Chart.image.histogram(changeImg.select('dur'), aoi,
500, 20, 1);
print(magOutputHist);
print(durOutputHist);
```

9. Now let's display the maps. Note that we're scaling the magnitude map on the fly when we're displaying it so that the output is actually in units of NDVI (and not NDVI *1000). We're also using some handy pre-defined colour ramps that are available on gitHub.

```
//Library taken from here https://github.com/gee-community/ee-palettes
var palettes = require('users/gena/packages:palettes');
var paletteRamp = palettes.colorbrewer.RdYlBu[7];


// set visualisation parameters (based on a predefined palettes)
var yodVis = {min:2012, max:2016, palette:paletteRamp}; // Colour ramp red
- yellow - blue
var magVis = {min:0.3, max:1, palette:paletteRamp}; //Colour ramp red -
yellow - blue
var durVis = {min:1, max: 5, palette:paletteRamp}; //Colour ramp red -
yellow - blue


//Display the change attribute maps
Map.centerObject(aoi, 10);
Map.addLayer(changeImg.select('mag').multiply(0.001), magVis, 'Magnitude of
Change'); //scaled so that it's easier to understand
Map.addLayer(changeImg.select('yod'), yodVis, 'Year of Detection');
Map.addLayer(changeImg.select('dur'), durVis, 'Duration of Change');

// Note that there are other layers which we are not mapping. These will be
listed in the right hand console windows
print(changeImg, "Change image layers");
```

10. The final step is to export the imagery to your Google Drive. Here, we will output just the magnitude and year of detection image as an example but you can select any image output that you want (see the “bands” listed when you printed the image to the console to check that you have the correct names!)

```
//Export change data
// Example single band image

var exportMagImg =
changeImg.select('mag').multiply(0.001).unmask(0).float(); //Set masked
value to 0
```

```
Export.image.toDrive({
  image: exportMagImg, //The name of your image
  description: 'KingFire_Magnitude_map',
  folder: 'KingFire_Output',
  fileNamePrefix: 'KingFire_Magnitude_map',
  region: aoi,
  scale: 30,
  maxPixels: 1e13
});

var exportYodImg = changeImg.select(['yod']).unmask(0).short(); //Set
masked value to 0 and output as integer (i.e. short)

Export.image.toDrive({
  image: exportYodImg, //The name of your image
  description: 'KingFire_YOD_map',
  folder: 'KingFire_Output',
  fileNamePrefix: 'KingFire_YOD_map',
  region: aoi,
  scale: 30,
  maxPixels: 1e13
});
```

After running the above code don't forget to go to the **Tasks** tab and click **Run** to begin exporting the disturbance maps to your GoogleDrive account. This may take a while to process so please be patient. Remember that you can close GEE and the process will still continue.

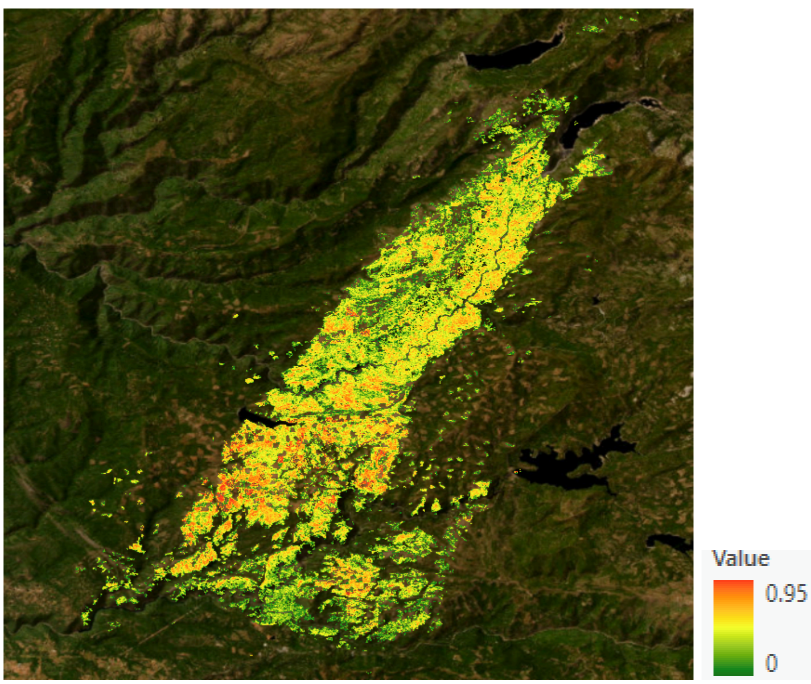
When ready, download them from GoogleDrive and try opening them in ArcGIS Pro/QGIS to visualise them (see video on the BB site if you are not sure how to). *Note: Remember that when you export the data in this way, you don't export any of the visualisation parameters, just the data themselves so don't stress over those visualisation parameters! You  need to customise how the data look in the package that you are importing the data into. You can export the data "with" the GEE visualisation (see GEE help) but these data can not be analysed further, they are just "pictures" for show!*

# Task 3: Exploring the outputs from LandTrendr

Now that the data have been generated, it's important to be able to interpret the outputs and undertake further analysis where required and/or use the images to help refine your choice of LandTrendr parameters. Often it is quicker to do this on your local machine using a GIS software (you can also upload the tif files back into GEE as an asset and view them that way - this will prevent you having to re-run theLandTrendr code if you want to do further analysis in GEE!) .

1. Open the **magnitude of the change** layer in ArcGIS Pro/QGIS to visualise it.

   For example, I've quickly used a colour ramp to show the magnitude of disturbance (i.e. change in NDVI) in the image below as this layer has continuous values.




2. Have a go at loading your own data into a GIS and adjusting the symbology to help you interpret the outputs.

Once you've made it this far you may want to do some additional analyses on these output data to help ***quantify*** the spatial patterns - especially if you are thinking of LandTrendr as the main methodology that you will use in assessment 2. For example you may want to compare rates of vegetation loss during the disturbance event vs rates of vegetation gain after the event (hint: you may want to use some of your layers as a mask!), or you may want to look at disturbance characteristics across different land cover types (i.e. combine the outputs with other data such as a land cover classification) or disturbance severities. Remember that the "story" of each pixel is reflected in the value for that pixel in each of the different LandTrendr output layers - all pixels in any given layer/image are not all from the "same" disturbance event - the **Year of detection image** is an important layer/output for identifying those different disturbance events. Consequently, when used and interpreted correctly, LandTrendr can tell you about spatial *and* temporal disturbance patterns affecting an area, depending on how you choose to combine the different output products!

There are plenty of options for further analyses, examples of which you will find in the published literature. Remember you can do these types of additional analyses both within and outside of GEE; it's entirely up to you how you want to further process the outputs from LandTrendr! For further help doing analyses outside of GEE you can consult the spatial analysis ArcGIS Pro help e.g.
https://pro.arcgis.com/en/pro-app/latest/tool-reference/spatial-analyst/an-overview-of-the-spatial-analyst-toolbox.htm

# Final thoughts

Remember that **adjusting the parameters in the GEE script really makes a difference to what the algorithm detects**, so it's essential to understand what each of them does and to think about your choices in relation to your study (i.e. what do you want to show?) and to clearly justify those decisions in any write up - you need to demonstrate a clear understanding of the approach. For your write up you should produce your final figures in a GIS package e.g. ArcGIS Pro or QGIS. **DO NOT SCREEN GRAB images from GEE**!

**Additional Resources**

- Introduction to the LT-GEE: https://emapr.github.io/LT-GEE/introduction.html
- LT-GEE explanation of each of features: https://emapr.github.io/LT-GEE/ui-applications.html
- LT-GEE Example Scripts: https://emapr.github.io/LT-GEE/example-scripts.html
- Cohen, W B., *et. al.*, (2017) How similar are forest disturbance maps derived from different Landsat time series algorithms? *Forests*, doi:10.3390/f8040098.
- Kennedy RE, Yang Z, Cohen WB (2010) Detecting trends in forest disturbance and recovery using yearly Landsat time series: 1. LandTrendr - Temporal segmentation algorithms. *Remote Sensing of Environment* 114:2897–2910. https://doi.org/10.1016/j.rse.2010.07.008
- Kennedy RE, Yang Z, Gorelick N, et al (2018) Implementation of the LandTrendr algorithm on Google Earth Engine. *Remote Sensing* 10:691. https://doi.org/10.3390/rs10050691
- Negrón JF and Cain, BC (2018), Mountain Pine Beetle in Colorado: A Story of Changing Forests, *Journal of Forestry*, doi: 10.1093/jofore/fvy032.


**END OF PRACTICAL**

# Assessment 2: Multi-temporal analysis – Short Research Project

The assessment is associated with the **Phenology and LandTrendr time series practicals** (i.e. practicals 4 and 5) and is worth **70% of the overall mark**. The **word limit** for this assessment is **2,500 words.** This includes all text but excludes the reference list, numbers in tables and text that is included in a figure (excluding figure captions). Everything else is included.

## Question

Each student is required to undertake a short independent research project, involving the use of multi-temporal (*i.e. time-series*) data. Specifically, projects **must "demonstrate" how multi-temporal satellite data can play a key role in continuous monitoring of the Earth's surface.** Projects can focus on any application of multi-temporal data analysis but projects should focus on the use of *continuous quantitative data* (i.e. using time-series data and *not performing a "change detection" that uses just two or three discrete or classified images*) and not directly replicate any of the practical exercises i.e. use the same data, for the same location.

**You must select one of the time-series analysis methods that we have used in class.** You may supplement either method with other methods if you so wish (and if it makes sense to do so), **but the basis of your study must be one of the methods that we used in class.**

The assessment is designed to:

- Provide hands-on experience in designing and implementing a remote sensing project.
- Allow you to explore in a formal manner, an application of remote sensing that interests you.
- Provide you with experience in writing a research paper, with a **focus on following the conventions of a journal article** with respect to:
    - Structure
    - Language

- References and citations
- Figures & Tables

You should choose the project based on the feasibility of carrying out the work in the given timescale – remember this is not a dissertation project!

## Write-up instructions

Write a research paper based on the research questions and methodology that you have decided upon, citing peer reviewed literature where relevant (i.e. not websites!). Include images, charts and tables depicting and quantifying the temporal/spatial trends, and any other images that you think are *relevant*. I would like to see at least one satellite image of the area for context!

Your research paper should be **structured as a journal style research paper** e.g. in the following way (***you do not need an abstract***):

1. **Introduction -** Should include (i) an introduction to the topic area which draws on published literature to help state the importance/relevance of the work and any gaps in the knowledge that it may contribute to filling, (ii) study location (although this can be moved to the methodology if it's more appropriate) and justification for this location (iii) clearly state *the overall aim and specific research objectives or questions, which the project aims to address*. It's often useful to also provide a few lines which outline "how" you plan to achieve those objectives or answer those questions i.e. what methodological approach. ***Note****: You should select a title for your paper, which represents what your study is about.*

2. **Methodology** - You should explain your entire methodology, including justifications for choices made, where appropriate. Avoid writing in the first person e.g. "I did x".. You should include information pertaining to the data that you used and how and why the data were analysed in a particular way. It's usually advisable to complement any text with flow charts, tables and/or diagrams. *Flowcharts should show the* ***flow of data between steps*** *including decisions and feedback loops (e.g. iterations, where relevant), how data were combined etc. and not just list the steps in a linear fashion – that is just a list!* You should also clearly state if/how you analysed the data outputs, including any statistical methods used, *clearly stating what their purpose is* - try to link specific sets of analyses with your specific research questions. Your

method should be sufficiently detailed so that it is reproducible, regardless of the software/platform used.

3. **Results** - This section should include your results. Do not just paste the images or figures into this section. The results section should include text highlighting interesting features that will be drawn upon in the discussion section. Always try to place the text that discusses the figure *before* the figure is displayed. **Please see any research journal article for advice on how to do this if you are not sure.** All figures and tables need to include a figure or table number and a full caption explaining what the figure shows. All figures need to be referred to from the text. Figure captions should be placed below the figure and table captions above the table. Think about how to create figures that are "reader friendly". Think about their position in the text (*usually placing figures after the text that refers to them is better than before*), axis labels, legends and text size - they must be legible!

**4. Discussion** – Here is where you will "discuss" and explain what your results show and put your findings *in context to other literature on the subject area*. You may also want to discuss possible "causes" of changes observed, even if you don't have a definitive answer! Use the literature to back up your key points/observations in this section.

4. **Conclusion and further work** - Summarise your main findings in relation to your research questions/objectives. Are there any improvements that could be made to your methodology or any alternative methods/approaches which you think might be enhance the study going forward i.e. What alternative or additional data could you use and what methods/approaches could you try and why?

5. **References** - A complete and correctly formatted reference list

6. **Link** to you GEE code

The final report is to be submitted as a single PDF document via the BB site.

## Advice!

Helpful steps:

1. Think of an area(s) which is (are) likely to have experienced observable land cover change within the lifetime of a sensor of your choice. Datasets from Landsat or MODIS are the most commonly used at the moment.

2. Generate an overall aim and a series of research questions (*not methodological steps!)* for your project. You will need to pose the work as a valid research question (s). You can construct your research question(s) as you see fit as long as they relate to quantitative land cover change. Looking at the literature will help you to decide on a location(s) and justify the research question(s) that you develop.

3. Undertake the analysis using the approach of your choice**.** Do not just create land cover classification maps for discrete points in time; unless they are used to "supplement" your main methodology. You do not need to use "all" tools and methods that we have covered. Part of the assessment is to show that you are able to select an "appropriate" methodology so remember to *justify your methodology* and choices made (*based on the published scientific literature)* in the methods section.

4. Remember to try and both *visualise* and *quantify* how the environment has changed over time within your defined area(s) of interest. A good way to determine "how" to present your data is by reading journal articles which present time-series results using your chosen method. You will receive extra credit for more challenging analysis, but the execution and interpretation must also be correct.

5. Start early! - I can only provide help during working hours and I cannot provide coding help less than 48 hrs before the submission deadline. Please use the discussion board to ask questions as opposed to e-mail and remember to help each other too!

**Some example titles of previous submissions**

- Monitoring the change of a forest ecosystem to changing environmental conditions: The Paphos forest as a case study
- Desertification monitoring in xxx
- Assess wildfire disturbance and recovery in location x using the LandTrendr algorithm

- Tracking forest mortality from insect pests
- Climate induced vegetation disturbances in xxx
- Spatial and temporal patterns of forest disturbance and regrowth in xxx
- Urban vs rural land surface phenology

**Recent papers that use multi-temporal satellite data (there are many many more!)**


- Bost, D.S., Reilly, M.J., Jules, E.S., DeSiervo, M.H., Yang, Z. and Butz, R.J., 2019. Assessing spatial and temporal patterns of canopy decline across a diverse montane landscape in the Klamath Mountains, CA, USA using a 30-year Landsat time series. *Landscape Ecology*, *34*(11), pp.2599-2614.
- Bright, B.C., Hudak, A.T., Kennedy, R.E., Braaten, J.D. and Khalyani, A.H., 2019. Examining post-fire vegetation recovery with Landsat time series analysis in three western North American forest types. *Fire Ecology*, *15*(1), p.8.
- Du, J., Li, K., He, Z., Chen, L., Lin, P. and Zhu, X., 2019. Daily minimum temperature and precipitation control on spring phenology in arid-mountain ecosystems in China. *International Journal of Climatology*.
- Jenkins, L.K., Barry, T., Bosse, K.R., Currie, W.S., Christensen, T., Longan, S., Shuchman, R.A., Tanzer, D. and Taylor, J.J., 2019. Satellite-based decadal change assessments of pan-Arctic environments. *Ambio*, pp.1-13.
- Kato, A., Carlson, K.M. and Miura, T., 2021. Assessing the inter-annual variability of vegetation phenological events observed from satellite vegetation index time series in dryland sites. *Ecological Indicators*, *130*, p.108042.
- Kennedy, R.E., Yang, Z., Cohen, W.B., Pfaff, E., Braaten, J. and Nelson, P., 2012. Spatial and temporal patterns of forest disturbance and regrowth within the area of the Northwest Forest Plan. *Remote Sensing of Environment*, *122*, pp.117-133.
- McIntosh, M.M., Holechek, J.L., Spiegal, S.A., Cibils, A.F. and Estell, R.E., 2019. Long-Term Declining Trends in Chihuahuan Desert Forage Production in Relation to Precipitation and Ambient Temperature. *Rangeland ecology & management*, *72*(6), pp.976-987.
- Testa, S., K. Soudani, L. Boschetti, and E. Borgogno-Mondino. 2018. MODIS-derived EVI, NDVI and WDRVI Time Series to Estimate Phenological Metrics in French Deciduous Forests. *International Journal of Applied Earth Observation and Geoinformation* 64: 132–144. doi:10.1016/j.jag.2017.08.006.
- Touhami, I., Moutahir, H., Assoul, D., Bergaoui, K., Aouinti, H., Bellot, J. and Andreu, J.M., 2022. Multi-year monitoring land surface phenology in relation to climatic variables using MODIS-NDVI time-series in Mediterranean forest, Northeast Tunisia. *Acta Oecologica*, *114*, p.103804.
- Walker, J.J., De Beurs, K.M. and Henebry, G.M., 2015. Land surface phenology along urban to rural gradients in the US Great Plains. *Remote Sensing of Environment*, *165*, pp.42-52.
- Xu, H., Wei, Y., Liu, C., Li, X. and Fang, H., 2019. A Scheme for the Long-Term Monitoring of Impervious− Relevant Land Disturbances Using High Frequency Landsat Archives and the Google Earth Engine. *Remote Sensing*, *11*(16), p.1891.
- Zhu, L., Liu, X., Wu, L., Tang, Y. and Meng, Y., 2019. Long-Term Monitoring of Cropland Change near Dongting Lake, China, Using the LandTrendr Algorithm with Landsat Imagery. *Remote Sensing*, *11*(10), p.1234.