

Εθνικό Μετσόβειο Πολυτεχνείο

Διατμηματικό Πρόγραμμα Μεταπτυχιακών Σπουδών

Επιστήμης Δεδομένων και Μηχανικής Μάθησης



Πρώτη Εργαστηριακή Άσκηση στα πλαίσια του μαθήματος
Ανάλυση και Επεξεργασία Γεωχωρικών Δεδομένων

*Ανάκτηση και Ανάλυση Δεδομένων, Χρονοσειρών
και Γεωχωρικών Προϊόντων
από Επιχειρησιακά Πληροφοριακά Συστήματα και
Υπηρεσίες*

Συγγραφή: Χρήστος Νίκου
ΑΜ: 03400146

EMAIL:CHRISTOSNIKOU@MAIL.NTUA.GR

1 Μαΐου 2022

Δομή της τεχνικής έκθεσης

Σε αυτό το σημείο δίνουμε μια σύντομη περιγραφή της δομής της παρούσας τεχνικής έκθεσης. Η έκθεση χωρίζεται σε δύο μέρη: 1) Στα βήματα και 2) στα ζητούμενα. Στο πρώτο κομμάτι υλοποιούμε τα βήματα της άσκησης και δίνουμε αναλυτικές επεξηγήσεις για τη διαδικασία υλοποίησης των βημάτων, επιπροσθέτως, παραθέτουμε σε ειδικό πλαίσιο κειμένου (*highlighted text*) τον κώδικα που χρησιμοποιήθηκε για το εκάστοτε βήμα. Στο δεύτερο μέρος της, έχοντας αναπτύξει τον απαιτούμενο κώδικα, απαντάμε στα ζητούμενα της άσκησης και παραθέτουμε τα αποτελέσματα για κάθε ένα απ' τα ερωτήματα. Τέλος, να αναφέρουμε ότι η συγκεκριμένη εργασία υλοποιήθηκε στο περιβάλλον του [Google Earth Engine](#) (GEE). Στα παρακάτω [links](#) μπορεί να βρεθεί και όλος ο κώδικας που αναπτύχθηκε στα πλαίσια της εργασίας στο περιβάλλον του GEE.

- <https://code.earthengine.google.com/41ccab0bec3e2776b339783feb8beaf4> - Βήματα Part A.
- <https://code.earthengine.google.com/7a879d96dda9dc6adce27dd5fd65d876> - Βήματα Part B.
- <https://code.earthengine.google.com/eb4c25c1451e938199ac003192709e91> - Τα ζητούμενα.
- <https://code.earthengine.google.com/783720c3a4d82943d5b915071640e9c3> - Source code App.
- <https://chrisnick92.users.earthengine.app/view/app1> - To App.

Πρώτο Μέρος - Τα Βήματα

Βήμα 1

Εκφώνηση Δημιουργήστε ένα πολύγωνο σε μια περιοχή της Ελλάδας (έκταση δημοτικού διαμερίσματος) με καλλιέργειες (π.χ. κάμπος Λάρισας) και ανακτήστε μια χρονοσειρά L8 (collection) για το παραπάνω πολύγωνο με όλα τα δεδομένα για όλο το έτος 2019. Τα δεδομένα της χρονοσειράς θα πρέπει να έχουν νεφοκάλυψη μικρότερη του 20%. Οπτικοποιήστε στην κονσόλα τα αποτελέσματα (print, JSON, metadata, etc.)

Σε αυτό το σημείο επιλέγουμε μέσω της κονσόλας του GEE την περιοχή ενδιαφέροντος (πολύγωνο) που θέλουμε να μελετήσουμε. Το πολύγωνο που επιλέγουμε αντιστοιχεί σε αγροτικές καλλιέργειες που βρίσκονται έξω απ' την πόλη της Λάρισας. Στην παρακάτω εικόνα βλέπουμε την ευρύτερη περιοχή της Λάρισας και το επιλεγμένο πολύγωνο.



Σχήμα 1: Η περιοχή ενδιαφέροντος έξω απ' την πόλη της Λάρισας.

Στη συνέχεια φορτώνουμε τα δεδομένα του δορυφόρου [Landsat 8](#) και κρατάμε όλες τις φωτογραφίες

για το 2019 που περιέχουν το πολύγωνο με νεφοκάλυψη μικρότερη του 20%. Παρακάτω βλέπουμε και τις αντίστοιχες εντολές οι οποίες υλοποιούν αυτό το ερώτημα.

```
var L8 = ee.ImageCollection("LANDSAT/LC08/C01/T1");  
// Keep data with cloud cover <= 20 for year 2019  
var unclouded_data = L8.filter(ee.Filter.lt('CLOUD_COVER', 20)).  
filter(ee.Filter.date('2019-01-01', '2019-12-31'));  
// Step 3 - Keep data only on ROI  
var unclouded_data_roi = unclouded_data.filter(ee.Filter.bounds(RuralPoly));  
print('The total number of images on roi is', unclouded_data_roi.size());  
print('The JSON file of roi is the following', unclouded_data_roi);
```

Βήμα 2

Εκφώνηση Από την παραπάνω χρονοσειρά εικόνων, εξάγετε την εικόνα με την μικρότερη νεφοκάλυψη στην περιοχή. Οπτικοποιήστε την εικόνα αυτή στην κονσόλα: προσθέστε στο χάρτη του Code Editor ένα φυσικό έγχρωμο σύνθετο L8 (RGB-432) και ένα ψευδέγχρωμο σύνθετο (RGB-543). Για την ίδια εικόνα, υπολογίστε τον δείκτη βλάστησης NDVI (float) και απεικονίστε τον στον χάρτη του Code Editor, χρησιμοποιώντας παλέτα χρωμάτων (από κόκκινο σε πράσινο).

Εν συνεχεία με τις εντολές

```
// RGB visualization  
var RGB_viz = {min: '6500', max: '11000', bands: ['B4', 'B3', 'B2']};  
// Step 4 - Image with lowest cloud cover  
var unclouded_image = ee.Image(unclouded_data_roi.sort('CLOUD_COVER').first());  
Map.addLayer(unclouded_image, RGB_viz, 'Clearest Img RGB');
```

ορίζουμε τη χρωματική παλέτα που θα χρησιμοποιούμε από δω και στο εξής για τη φυσική χρωματική απεικόνιση (RGB) και επιλέγουμε την εικόνα με το μικρότερο ποσοστό νεφοκάλυψης για την περιοχή ενδιαφέροντος. Τέλος, με την εντολή *Map.addLayer* προσθέτουμε αυτό το στρώμα στην εικόνα. Στην παρακάτω εικόνα βλέπουμε και το αποτέλεσμα που παίρνουμε.



Σχήμα 2: Η εικόνα με τη μικρότερη νεφοκάλυψη στο φυσικό έγχρωμο σύνθετο (RGB).

Για να εξάγουμε και την εικόνα με ένα ψευδέγχρωμο σύνθετο επιλέγουμε το κομμάτι που αντιστοιχεί στην εγγύς υπέρυθρη ακτινοβολία του ηλεκτρομαγνητικού φάσματος αντί για το κομμάτι του φάσματος που αντιστοιχεί στο ερυθρό χρώμα και εν συνεχεία προσαρμόζουμε αναλόγως τη χρωματική παλέτα. Παρακάτω βλέπουμε τις σχετικές εντολές και το αποτέλεσμα που παίρνουμε όταν προσθέσουμε αυτό το στρώμα στην εικόνα.

```
// False color visualization  
var Pseudo_viz = {min: '6500', max: '15000', bands: ['B5', 'B4', 'B3']};
```

```
Map.addLayer(unclouded_image, Pseudo_viz, 'Clearest Img Pseudo');
```



Σχήμα 3: Η εικόνα με τη μικρότερη νεφοκάλυψη στο ψευδέγχρωμο σύνθετο.

Για να υπολογίσουμε τον δείκτη βλάστησης (*NDVI*) πηγαίνουμε σε κάθε pixel της εικόνας και υπολογίζουμε τον λόγο

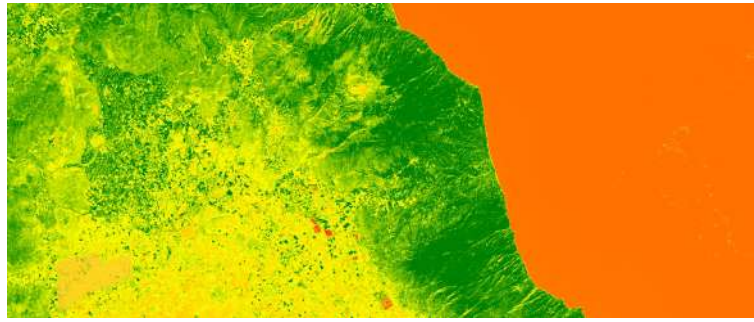
$$NDVI = \frac{NIR - Red}{NIR + Red},$$

όπου με *NIR*, *Red* συμβολίζουμε τις τιμές του εικονοστοιχείου στο κομμάτι του φάσματος που αντιστοιχεί στο εγγύς υπέρυθρο και στο ερυθρό αντιστοιχώς. Ο λόγος για τον οποίο ο δείκτης ονομάζεται δείκτης βλάστησης είναι γιατί η χρωστική των φυτών, η χλωροφύλλη, απορροφά σε υψηλά ποσοστά το ορατό φως κατά τη διάρκεια της φωτοσύνθεσης. Από την άλλη πλευρά η κυτταρική δομή των φύλλων ανακλά σε μεγάλο βαθμό το εγγύς υπέρυθρο φως. Εάν η ανακλώμενη ακτινοβολία στο εγγύς υπέρυθρο μήκος κύματος είναι πολύ περισσότερη από ότι στο ορατό μήκος φωτός, τότε η βλάστηση στο συγκεκριμένο εικονοστοιχείο είναι πιθανό να είναι πυκνή και ενδέχεται να αποτελείται από κάποιο είδος δάσους. Εάν υπάρχει πολύ μικρή διαφορά στην ένταση του ανακλώμενου ορατού και εγγύς υπέρυθρου μήκους φωτός, τότε η βλάστηση είναι ενδεχομένως αραιή και ενδέχεται να αποτελείται από ποολίβαδα, τούνδρα ή έρημο (πηγή: https://www.esa.int/SPECIALS/Eduspace_Global_GR/SEMRTITWT1H_0.html). Έτσι, υψηλές τιμές (κοινά στο 1) του δείκτη βλάστησης αντιστοιχούν σε περιοχές με έντονη βλάστηση ενώ χαμηλές τιμές σε περιοχές με λίγη βλάστηση έως καθόλου βλάστηση.

Για να εξάγουμε τον δείκτη βλάστησης σε κάθε εικονοστοιχείο για την εικόνα με τη χαμηλότερη νεφοκάλυψη κατασκευάζουμε πρώτα μια συνάρτηση *addNDVI* η οποία δέχεται μια εικόνα και την επιστρέφει έχοντας προσθέσει και τον δείκτη *NDVI* στις υπόλοιπες μπάντες. Παρακάτω βλέπουμε το σώμα της συνάρτησης, την εφαρμογή της στην εικόνα χαμηλότερης νεφοκάλυψης και το αντίστοιχο αποτέλεσμα.

```
var NDVI_viz = {min: '-0.1', max: '0.35', bands: 'NDVI',
palette: 'orangered, yellow, green'}; // NDVI visualization
// Function to calculate NDVI

var addNDVI = function(image){
  // NDVI = (NIR - Red) / (NIR + Red)
  var ndvi = image.normalizedDifference(['B5', 'B4']).float().rename('NDVI');
  return image.addBands(ndvi);
};
var unclouded_image = addNDVI(unclouded_image);
Map.addLayer(unclouded_image, NDVI_viz, 'Clearest Img NDVI');
```



Σχήμα 4: Η αναπαράσταση της εικόνας μέσω του NDVI.

Βήμα 3

Εκφώνηση Με βάση την παραπάνω χρονοσειρά των εικόνων L8 του 2019, δημιουργήστε νέες υπολογίζοντας για όλες τις εικόνες τους δείκτες NDVI, EVI, και NDWI (ονομάστε αντίστοιχα 'NDVI', 'EVI', 'NDWI', όλα float) για όλες τις διαθέσιμες χρονικές στιγμές.

Σε αυτό το βήμα κρατάμε ολόκληρη τη συλλογή εικόνων του δορυφόρου που περιέχουν τη περιοχή ενδιαφέροντος και υπολογίζουμε εκ νέου τον δείκτη βλάστησης NDVI, τον κανονικοποιημένο δείκτη νερού NDWI και τον ενισχυμένο δείκτη βλάστησης EVI για όλες τις εικόνες της συλλογής. Ο δείκτης νερού δίνεται μέσω της

$$NDWI = \frac{NIR - SWIR}{NIR + SWIR}$$

, όπου NIR είναι η τιμή του εικονοστοιχείου στη περιοχή του υπέρυθρου και SWIR η τιμή στο κοντινό υπέρυθρο (*Short wavelength infrared*). Η αντανάκλαση του κοντινού υπέρυθρου δείχνει τις μεταβολές στη περιεκτικότητα νερού της βλάστησης ενώ το υπέρυθρο επηρεάζεται από την εσωτερική δομή του φύλλου όσον αφορά τη στεγνή ύλη που περιέχει το φύλλο αλλά όχι απ'την υγρή του ύλη. Έτσι, ο συνδυασμός των δύο παραπάνω ποσοτήτων «αφαιρεί» την πληροφορία που δίνουν οι μεταβολές στη στεγνή ύλη στο εσωτερικό του φύλλου κρατώντας μόνο την πληροφορία που δίνουν οι μεταβολές όσον αφορά την υγρή ύλη του φύλλου στο εσωτερικό. Επομένως, υψηλές τιμές του δείκτη NDWI δείχνουν μεγάλη περιεκτικότητα νερού στο εσωτερικό του φύλλου ενώ αντίθετα μικρές τιμές του δείκτη δείχνουν χαμηλή περιεκτικότητα νερού (εώς και καθόλου) στο εσωτερικό του φύλλου. Ο δείκτης ενισχυμένης βλάστησης υπολογίζεται μέσω της

$$EVI = G \times \frac{(NIR - Red)}{(Nir + C_1 \times Red - C_2 \times Blue + L)},$$

όπου οι τιμές των σταθερών για τον Landsat 8 είναι οι $G = 2.5$, $C_1 = 6$, $C_2 = 7.5$ και $L = 1$.

Για τον υπολογισμό των παραπάνω δεικτών κατασκευάζουμε μια ενιαία συνάρτηση η οποία δέχεται ως είσοδο μια εικόνα και επιστρέφει ως έξοδο την ίδια εικόνα έχοντας προσθέσει επιπλέον τις μπάντες των τριών παραπάνω δεικτών. Παρακάτω βλέπουμε το σώμα της συνάρτησης, τον υπολογισμό των δεικτών σε όλη τη συλλογή των εικόνων του Landsat 8 για το 2019 που περιέχουν τη περιοχή ενδιαφέροντος καθώς και τις αντίστοιχες εικόνες για κάθε έναν από τους τρεις δείκτες ξεχωριστά.

```
var NDWI_viz = {min : '-0.15', max: '0.25', bands : 'NDWI', palette: 'orangered,yellow,green,lightblue,blue'};
```



```

var EVI_viz = {min: '-3', max: '1', bands: 'EVI', palette: 'brown, yellow, green'};

// A function to calculate all indices NDVI, NDWI, EVI
var calculate_indices = function(image){
  // EVI = 2.5*(NIR - Red) / (NIR + 6*Red - 7.5*Blue + 1)
  var nir = image.select('B5');
  var red = image.select('B4');
  var blue = image.select('B2');
  var evi = (nir.multiply(2.5).subtract(red.multiply(2.5)).divide(nir.add(red.multiply(6)).add(blue.multiply(-7.5)).add(1))).rename('EVI'));

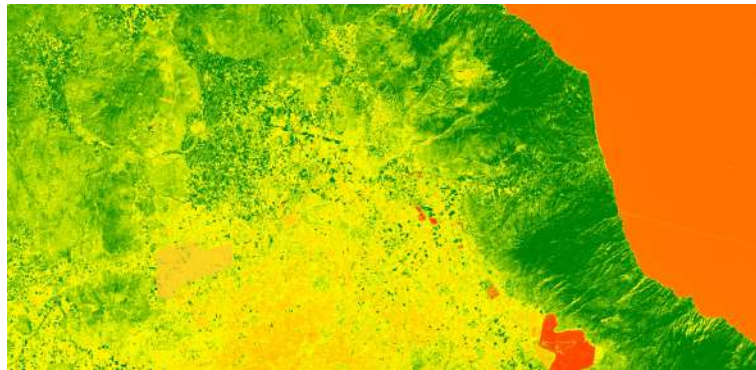
  var ndvi = image.normalizedDifference(['B5', 'B4']).float().rename('NDVI');
  var ndwi = image.normalizedDifference(['B5', 'B6']).rename('NDWI');

  return image.addBands(ndvi).addBands(ndwi).addBands(evi);
};
var unclouded_data_roi = unclouded_data_roi.map(calculate_indices);

// Adding Layers with all indices NDVI, NDWI, EVI
Map.addLayer(unclouded_data_roi.select('NDWI'), NDWI_viz, 'NDWI for year 2019');
Map.addLayer(unclouded_data_roi.select('NDVI'), NDVI_viz, 'NDVI for year 2019');
Map.addLayer(unclouded_data_roi.select('EVI'), EVI_viz, 'EVI for year 2019');

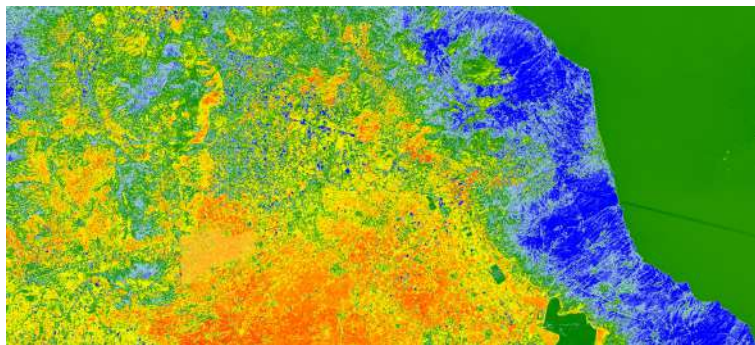
```

Στην επόμενη εικόνα βλέπουμε την αναπαράσταση του δείκτη βλάστησης *NDVI*. Όπως βλέπουμε και απ'τη χρωματική παλέτα που έχει επιλεχθεί για το συγκεκριμένο δείκτη, χαμηλές τιμές απεικονίζονται με κόκκινο-πορτοκαλί χρώμα (από -1-0.05), ενδιάμεσες τιμές (0.05-0.20) με κίτρινο και υψηλές τιμές (0.20-1) με πράσινο χρώμα.



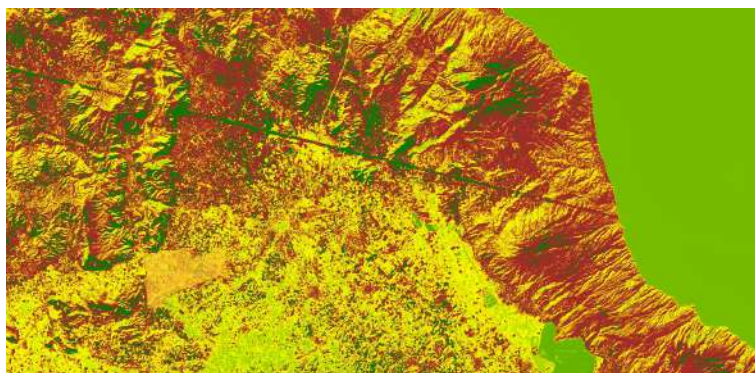
Σχήμα 5: Η αναπαράσταση του *NDVI* για όλη τη συλλογή εικόνων του 2019.

Παρακάτω βλέπουμε και την αντίστοιχη εικόνα για τον κανονικοποιημένο δείκτη νερού *NDWI*. Οι τιμές απεικονίζονται κατά άύξουσα σειρά με τα εξής χρώματα: πορτοκαλί-κόκκινο, κίτρινο, πράσινο, ανοικτό μπλε και μπλε.



Σχήμα 6: Η αναπαράσταση του NDWI για όλη τη συλλογή εικόνων του 2019.

Τέλος, βλέπουμε την εικόνα που αντιστοιχεί στον ενισχυμένο δείκτη βλάστησης EVI. Χαμηλές τιμές απεικονίζονται με καφέ χρώμα, ενδιάμεσες με κίτρινο και υψηλές τιμές με πράσινο.



Σχήμα 7: Η αναπαράσταση του EVI για όλη τη συλλογή εικόνων του 2019.

Βήμα 4

Εκφώνηση Στις παραπάνω χρονοσειρές (των 3 δεικτών βλάστησης) υπολογίστε για κάθε εικονοστοιχείο ποια ήταν η μέγιστη τιμή του δείκτη NDVI για το 2019 (*temporal features*). Στη συνέχεια υπολογίστε ποια ήταν η συγκεκριμένη μέρα του χρόνου (*day-of-year, DOY*) για την οποία το 2019 εμφανίστηκε η μέγιστη τιμή του NDVI. Αποθηκεύστε τις ημέρες του χρόνου (*DOY*), με τις μεγαλύτερες τιμές NDVI που υπολογίσατε, σε μία νέα εικόνα ίδιων διαστάσεων και απεικονίστε την στο χάρτη (χρωματική παλέτα της επιλογής σας).

Για να βρούμε τη μέγιστη τιμή του NDVI για το 2019 πηγαίνουμε σε κάθε εικονοστοιχείο και απ'όλη τη συλλογή των εικόνων βρίσκουμε τη μέγιστη τιμή ως προς τον δείκτη βλάστησης. Εν συνεχεία, συνθέτουμε μια εικόνα με τις μέγιστες τιμές του δείκτη βλάστησης για κάθε εικονοστοιχείο. Παρακάτω βλέπουμε τις εντολές γι' αυτό το ερώτημα και την εικόνα που προκύπτει.

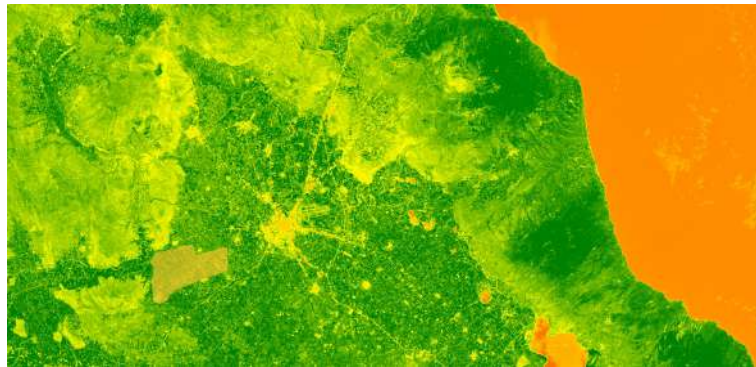
```
var NDVI_greenest_viz = {min: '-0.15', max: '0.55', bands: 'NDVI',
palette: 'orangered, yellow, green'};
var Season_viz = {min: '0', max: '366', bands: 'doy',
palette: 'lightblue, green, F37637, orangered'
};
```

```
// Step 6 - Greenest Pixels + Day of Year
```

```
//DOY for max ndvi
var addDOY = function(image){
  var img_date = ee.Date(image.date())
  var img_doy = ee.Number.parse(img_date.format('D'))
  return image.addBands(ee.Image(img_doy).rename('doy').toInt())
};

var greenest_pixels = unclouded_data_roi.map(addDOY).qualityMosaic('NDVI').select('NDVI', 'doy');

Map.addLayer(greenest_pixels, NDVI_greenest_viz, 'Greenest pixels for year 2019');
var seasonality_mapping = greenest_pixels.select('doy');
Map.addLayer(seasonality_mapping, Season_viz, 'Seasonality Mapping');
```



Σχήμα 8: Οι μέγιστες τιμές του δείκτη βλάστησης για το 2019.

Όπως βλέπουμε απ' την παραπάνω εικόνα η περιοχή του πολυγώνου, σε αντίθεση με την εικόνα του Σχήματος 5, απεικονίζεται με πράσινο χρώμα πράγμα το οποίο δείχνει ότι υπάρχουν περίοδοι που οι τιμές της βλάστησης σε αυτή την περιοχή είναι υψηλές. Έχοντας υπολογίσει και την κάθε ημέρα που σημειώθηκαν οι μέγιστες τιμές του δείκτη *NDVI* μπορούμε χρησιμοποιώντας κατάλληλη χρωματική παλέτα να απεικονίσουμε τα εικονοστοιχεία ανάλογα με την ημέρα που σημειώθηκαν οι μέγιστες τιμές (*Seasonality Mapping*). Στο παρακάτω σχήμα βλέπουμε αυτή την εικόνα. Εικονοστοιχεία με χρώμα ανοικτό μπλε αντιστοιχούν στο πρώτο τρίμηνο του 2019 (Χειμώνας), πράσινο χρώμα στο δεύτερο τρίμηνο (Άνοιξη), καφέ χρώμα στο τρίτο τρίμηνο (Καλοκαίρι) και κόκκινο-πορτοκαλί στο τέταρτο τρίμηνο (Φθινόπωρο).



Σχήμα 9: Οι μέγιστες τιμές του δείκτη βλάστησης για το 2019.

Όπως βλέπουμε η έντονη βλάστηση στην περιοχή του πολυγώνου σημειώνεται μεταξύ του δεύτερου

(Άνοιξη) και τρίτου (Καλοκαίρι) τριμήνου του 2019, αποτέλεσμα το οποίο είναι αναμενόμενο αφού το πολύγωνο αντιστοιχεί σε περιοχή αγροτικών καλλιεργειών.

Βήμα 5

Εκφώνηση Στη συνέχεια δημιουργήστε μια νέα μεγαλύτερη χρονοσειρά δεδομένων L8 (όλα τα διαθέσιμα δεδομένα L8 μέχρι και σήμερα), για το ίδιο πολύγωνο και με ποσοστό νεφοκάλυψης μικρότερο του 20%. Υπολογίστε για την παραπάνω χρονοσειρά το δείκτη βλάστησης NDVI (ή άλλο φασματικό δείκτη) για όλα τα διαθέσιμα L8. Με βάση τεχνικές ανάλυσης χρονοσειρών, περιοδικότητας και αρμονικής μοντελοποίησης, πειραματιστείτε και προσαρμόστε βέλτιστα καμπύλες στα δεδομένα σας, δημιουργώντας συνθετικές χρονοσειρές. Επιλέξτε διαφορετικά (σχετικά μικρά σε μέγεθος και φασματικά συμπαγή) πολύγωνα με διαφορετικές καλλιέργειες ή και διαφορετικές χρήσεις γης και δημιουργήστε για αυτά συνθετικές χρονοσειρές.

Σε αυτό το σημείο κρατάμε όλα τα διαθέσιμα δεδομένα L8 μέχρι και σήμερα. Αυτό το πετυχαίνουμε χωρίς να βάλουμε κάποιον περιορισμό ως προς την ημερομηνία.

```
var all_data_ndvi = L8.filter(ee.Filter.bounds(RuralPoly)).  
filter(ee.Filter.lt('CLOUD_COVER', 20)).map(addNDVI);
```

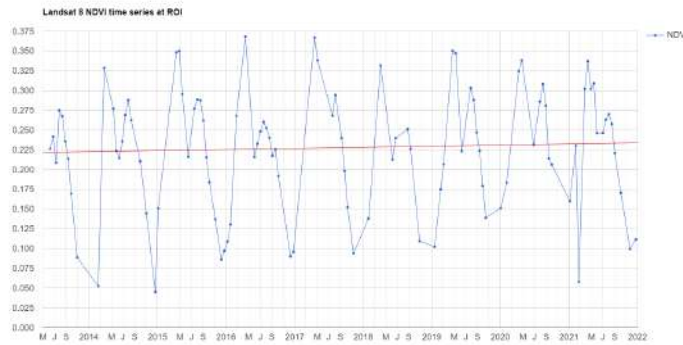
Λόγω της έλλειψης εικόνων για κάθε ημερομηνία, στο γράφημα συνάρτησης του δείκτη NDVI ως προς το χρόνο θα έχουμε και απουσιάζουσες τιμές. Για να έχουμε μια συνολική εικόνα της πορείας του δείκτη έχουν αναπτυχθεί διάφορες τεχνικές παραγωγής συνθετικών δεδομένων. Η πρώτη τεχνική αφορά τη γραμμική παλινδρόμηση. Δηλαδή, στην αναζήτηση της βέλτιστης ευθείας (ως προς το μέσο τετραγωνικό σφάλμα) που προσαρμόζεται στα δεδομένα μας. Πιο αναλυτικά, η υπόθεση που κάνουμε στη γραμμική παλινδρόμηση είναι ότι η τιμή του NDVI συσχετίζεται με το χρόνο μέσω της εξίσωσης

$$NDVI(t) = \beta_0 + \beta_1 \cdot t + \epsilon_t,$$

για κάποιες σταθερές β_0 , β_1 και ϵ_t εκφράζει το τυχαίο σφάλμα το οποίο υποθέτουμε πως ακολουθεί την κανονική κατανομή με μέση τιμή 0 και σταθερή διασπορά (υπόθεση ομοιοσκεδαστικότητας). Η εκτίμηση των σταθερών παραμέτρων γίνεται ελαχιστοποιώντας την ποσότητα

$$MSE(\beta_0, \beta_1) = \frac{1}{N} \sum_{i=1}^N \left(NDVI(t) - \beta_0 - \beta_1 \cdot t - \epsilon_t \right)^2,$$

ως προς β_0, β_1 . Στην παρακάτω εικόνα βλέπουμε το πως κυμαίνεται ο μέσος δείκτης βλάστησης του πολυγώνου στον χρόνο καθώς και την προσαρμοσμένη ευθεία.

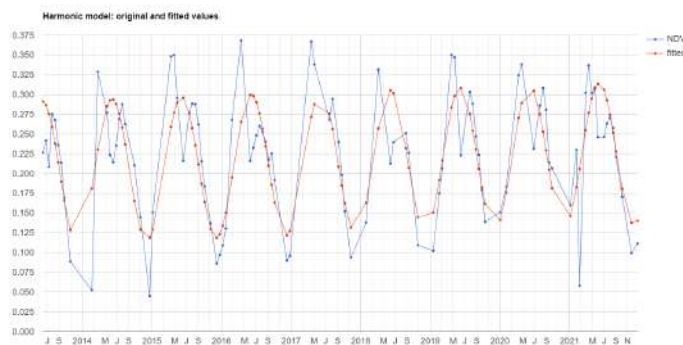


Σχήμα 10: Η συνθετική καμπύλη μέσω της γραμμικής παλινδρόμησης.

Η επόμενη τεχνική αφορά την αρμονική παλινδρόμηση. Όπως βλέπουμε και στο σχήμα 10, ο δείκτης βλάστησης NDVI φαίνεται να εμφανίζει κάποια περιοδική συμπεριφορά. Αυτό είναι αναμενόμενο αν λάβουμε υπόψιν ότι κατά τις μέρες της άνοιξης και του καλοκαιριού αναμένουμε να έχουμε υψηλές τιμές του δείκτη και αυτές να φθίνουν προς τον χειμώνα και το φθινόπωρο. Επομένως, είναι λογικό να υποθέσουμε κάποια ημιτονοειδή συνάρτηση για το δείκτη βλάστησης. Συνεπώς, κατά την αρμονική παλινδρόμηση υποθέτουμε ότι

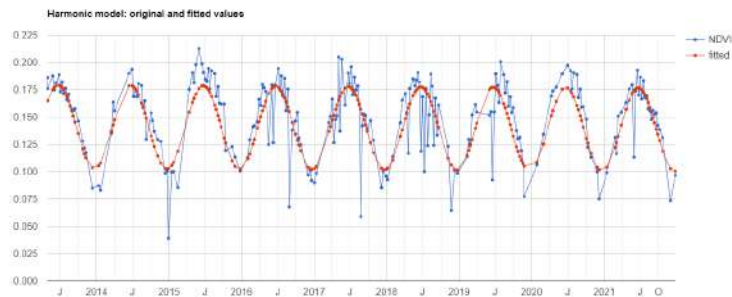
$$NDVI(t) = \beta_0 + \beta_1 \cdot t + \beta_2 \cdot \cos(2\pi\omega t) + \beta_3 \sin(2\pi\omega t) + \epsilon_t,$$

όπου ϵ_t είναι το τυχαίο σφάλμα για το οποίο υποθέτουμε ότι $\epsilon_t \sim \mathcal{N}(0, \sigma^2)$. Τώρα, οι συντελεστές προς εκτίμηση είναι οι $\beta_0, \beta_1, \beta_2$, οι οποίοι, όπως και στη περίπτωση της γραμμικής παλινδρόμησης, προσδιορίζονται ελαχιστοποιώντας το μέσο τετραγωνικό σφάλμα. Στο επόμενο γράφημα βλέπουμε την συνθετική καμπύλη που προκύπτει μέσω αυτής της μεθόδου.



Σχήμα 11: Η συνθετική καμπύλη μέσω της αρμονικής παλινδρόμησης.

Όπως βλέπουμε αυτή η μέθοδος περιγράφει πιο αποτελεσματικά την πορεία του δείκτη βλάστησης από την γραμμική παλινδρόμηση. Στο επόμενο γράφημα βλέπουμε και την πορεία του δείκτη βλάστησης σε συνάρτηση με το χρόνο για ένα τρίγωνο στο κέντρο της Αθήνας.



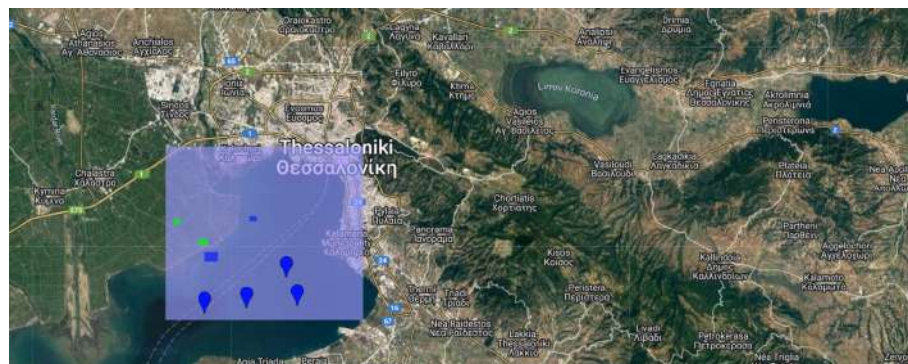
Σχήμα 12: Η συνθετική καμπύλη για ένα τρίγωνο στο κέντρο της Αθήνας.

Όπως βλέπουμε και στη περίπτωση του κέντρου της Αθήνας έχουμε περιοδική συμπεριφορά, η εμφανής διαφορά σε σχέση με το προηγούμενο γράφημα είναι ότι σε αυτή την περίπτωση οι τιμές του NDVI είναι μικρότερες όπως και περιμέναμε.

Βήμα 6

Εκφώνηση Δημιουργήστε ξανά ένα πολύγωνο σε μια περιοχή της Ελλάδας, το οποίο θα περιλαμβάνει διάφορες χρήσεις γης (νερό, βλάστηση, δρόμοι, αστικός ιστός) και ανακτήστε μία αντίστοιχη χρονοσειρά L8 με την προηγούμενη. Δημιουργήστε νέα layers με χαρακτηριστικά σημεία/πολύγωνα για κάθε διαφορετική χρήση γης που παρατηρείτε. Στη συνέχεια εκπαιδεύστε τον ταξινομητή CART χρησιμοποιώντας τα δεδομένα από τα layers των χρήσεων γης, και πραγματοποιήστε ταξινόμηση χρησιμοποιώντας το σύνολο των φασματικών καναλιών της χρονοσειράς. Χρησιμοποιήστε μπλε για το νερό, πράσινο για την βλάστηση, και κόκκινο για τον αστικό ιστό.

Σε αυτό το βήμα δημιουργούμε ένα πολύγωνο το οποίο αποτελείται κυρίως από τριων ειδών χρήσης γης, νερό, βλάστηση και αστικό ιστό. Επίσης, επιλέγουμε και και κάποια υποσύνολα του αρχικού μας πολυγώνου τα οποία θα αποτελέσουν τα δειγματοστοιχεία που θα εκπαιδεύσουμε τον ταξινομητή CART. Στην επόμενη εικόνα βλέπουμε το πολύγωνο καθώς και τα δειγματικά σύνολα.



Σχήμα 13: Το επιλεγμένο πολύγωνο για ταξινόμηση και τα δειγματοσύνολα.

Με τις παρακάτω εντολές δημιουργούμε τα δεδομένα L8 και απομονώνουμε την περιοχή ενδιαφέροντος. Επίσης, δημιουργούμε το σύνολο δεδομένων και εκπαιδεύουμε τον ταξινομητή.

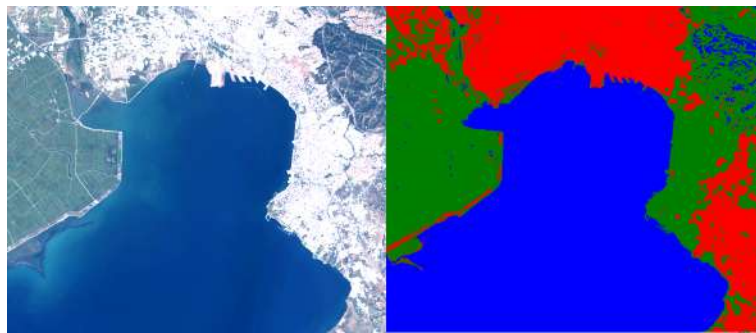
```

var RGB_viz = {min: '6500', max: '11000',
bands: ['B4', 'B3', 'B2']}; // RGB visualization
var all_data_L8 = L8.filter(ee.Filter.bounds(polygon2)).
filter(ee.Filter.lt('CLOUD_COVER', 20)).median().clip(polygon2);
Map.addLayer(all_data_L8, RGB_viz, 'Classification data - L8');
var features = all_data_L8.bandNames();
print('Features: ', features);
var training_pts = vegetation.merge(sea).merge(urban);
var training_data = all_data_L8.sampleRegions({collection: training_pts,
properties: ['class'], scale: 30
});

var classifier = ee.Classifier.smileCart();
var classifier = classifier.train({features: training_data,
classProperty: 'class', inputProperties: features});
var classified_image = all_data_L8.classify(classifier);
var Class_viz = {min: 0, max: 2, palette: ['green', 'blue', 'red']};
Map.addLayer(classified_image, Class_viz, 'Classified Image');

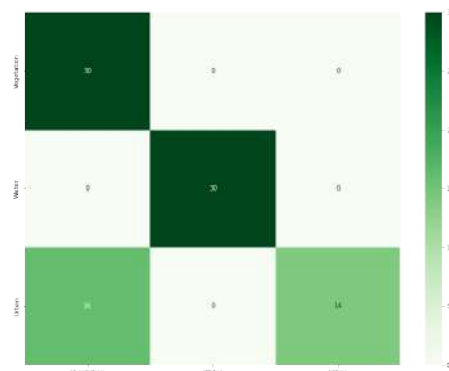
```

Στο επόμενο σχήμα βλέπουμε εποπτικά το αποτέλεσμα της ταξινόμησης μέσω της προσθήκης ενός layer όπου τα σημεία που αντιστοιχούν στο νερό απεικονίζονται με μπλε, τα σημεία που αντιστοιχούν σε βλάστηση με πράσινο και τα σημεία που αντιστοιχούν σε αστικό ιστό με κόκκινο χρώμα.



Σχήμα 14: Η πραγματική εικόνα σε σύγκριση με την ταξινομημένη εικόνα.

Για να αξιολογήσουμε καλύτερα την απόδοση του ταξινομητή, δημιουργούμε ένα σύνολο σημείων επικύρωσης. Δηλαδή, επιλέγουμε 30 σημεία απ' το κάθε είδος γης (βλάστηση, νερό, αστικό ιστό) και έχοντας εκπαιδεύσει τον ταξινομητή κατηγοριοποιούμε τα 30 σημεία και βλέπουμε το αποτέλεσμα.



Σχήμα 15: Ο πίνακας σύγχυσης της ταξινόμησης.

Δεύτερο Μέρος - Τα Ζητούμενα

Ζητούμενο 1

Εκφώνηση Να περιγράψετε τις βασικές λειτουργικότητες τεσσάρων (από τα παραπάνω 7) πληροφοριακών συστημάτων διαχείρισης και ανάλυσης δεδομένων και γεωχωρικών προϊόντων. Να αξιολογήσετε και συγκρίνετε τις υπηρεσίες τους σε σχέση με τη λειτουργικότητα, χρηστικότητα, διαθεσιμότητα δεδομένων και διεπαφή χρήστη.

Για την ανάλυση και την επεξεργασία γεωχωρικών δεδομένων έχουν αναπτυχθεί διάφορα επιχειρησιακά πληροφοριακά συστήματα και υπηρεσίες τα οποία αποθηκεύουν συστηματικά όλα τα ανοικτά διαθέσιμα δεδομένα Παρατήρησης Γης (Earth Observation data - EO data) καθώς και άλλα γεωχωρικά δεδομένα και προϊόντα. Η ανάγκη ανάπτυξης των συστημάτων και υπηρεσιών έχει προκύψει απ' την ανάγκη επεξεργασίας μεγάλων όγκων δεδομένων, τα οποία, ειδικά στην περίπτωση των γεωχωρικών δεδομένων, είναι της κλίμακας των Petabyte. Έτσι, οι υπηρεσίες αυτές παρέχουν στον χρήστη τον απαραίτητο αποθηκευτικό χώρο και την υπολογιστική ισχύ για την επεξεργασία αυτών των δεδομένων.

Μερικά από τα πιο διαδεδομένα επιχειρησιακά πληροφοριακά συστήματα (Data and Information Access Services - DIAS) Ευρωπαϊκής προέλευσης στο πλαίσιο του προγράμματος [Copernicus](#) είναι τα: 1.[CREODIAS](#) [3], 2.[sobloo](#) [8], 3.[WEkEO](#) [9], 4.[ONDA](#) [6], 5.[Mundi](#) [5]. Επίσης, εταιρείες όπως η [Google](#) και η [Amazon](#) [7] παρέχουν αντίστοιχες υπηρεσίες για δεδομένα παρατήρησης γης (τόσο ανοικτά όσο και εμπορικά) σε συνδυασμό με ισχυρά υπολογιστικά συστήματα υψηλών επιδόσεων. Στο πλαίσιο αυτού του ζητήματος θα σχολιάσουμε και θα αναλύσουμε τις βασικές λειτουργίες τεσσάρων απ' τα παραπάνω επιχειρησιακά πληροφοριακά συστήματα.

Google Earth Engine

Το *Google Earth Engine (GEE)* αποτελεί μια υπολογιστική διαδικτυακή πλατφόρμα η οποία δίνει τη δυνατότητα στους χρήστες να επεξεργάζονται και να οπτικοποιούν γεωχωρικά δεδομένα μεγάλης κλίμακας. Το *GEE* ξεκίνησε από την *Google* το 2010 ως ιδιόκτητο λογισμικό και σήμερα είναι διαθέσιμο με δωρεάν εγγραφή. Η συγκεκριμένη πλατφόρμα αξιοποιεί αρκετές τεχνολογίες που έχουν αναπτυχθεί από την *Google* όπως το *Borg*, διάφορες κατανεμημένες βάσεις (*Bigtable* και *Spanner*), το κατανεμημένο σύστημα αρχείων *Colossus* (Colossus distributed file system) καθώς και το σύστημα παράλληλης επεξεργασίας (parallel pipeline execution framework) *FlumeJava*.

Ένα από τα πλεονεκτήματα του *GEE* είναι ότι παρέχει στους χρήστες ένα προγραμματιστικό σύστημα διεπαφής (Application Programming Interface - API) βασισμένο στη γλώσσα προγραμματισμού JavaScript καθώς και ένα στη γλώσσα προγραμματισμού Python. Τα παραπάνω συστήματα διεπαφής διευκολύνουν αρκετά την επεξεργασία και ανάλυση των δεδομένων χωρίς να χρειάζεται κάποια εξειδικευμένη γνώση για τη χρήση του υπολογιστικού περιβάλλοντος αφού οι βασικές διεργασίες μεταφράζονται μέσω των δύο παραπάνω υψηλών επιπέδου γλωσσών προγραμματισμού. Η πλατφόρμα παρέχει και έναν κατάλογο στον οποίο βρίσκονται όλα τα διαθέσιμα παρατηρησιακά δεδομένα όπως τις εικόνες διαφόρων δορυφόρων, μεταβλητές συσχετιζόμενες με το περιβάλλον, τον καιρό, του κλίματος κ.α.

Το *GEE* χρησιμοποιεί τέσσερα βασικά αντικείμενα για την αναπαράσταση των δεδομένων τα

οποία μπορούν να επεξεργαστούν μέσω του API του. Τα αντικείμενα τύπου Image αναπαριστούν τα raster δεδομένα τα οποία αποτελούνται από ένα ή περισσότερα bands, τα οποία περιέχουν το όνομα, τον τύπο δεδομένων και την κλίμακα του band. Το αντικείμενο τύπου ImageCollection περιέχει μια στοίβα ή μια χρονοσειρά εικόνων από κάποιον δορυφόρο. Τα vector δεδομένα αναπαριστώνται μέσω του τύπου αντικειμένου Feature. Αυτός ο τύπος αντικειμένου αντιστοιχεί σε κάποια γεωμετρία του χάρτη (σημείο, γραμμή ή πολύγωνο) μαζί με μια λίστα από ιδιότητες της κάθε γεωμετρίας. Ο τύπος αντικειμένου FeatureCollection αποτελεί μια ομαδοποίηση αντικειμένων τύπου Feature και παρέχει συναρτήσεις για την επεξεργασία δεδομένων, όπως ταξινόμηση, φιλτράρισμα και οπτικοποίηση. Ένα άλλο εξίσου σημαντικό χαρακτηριστικό του GEE είναι η δυνατότητα και ο άμεσος τρόπος που προσφέρει στους χρήστες να διαμειράζονται και να αξιολογούν τον ανεπτυγμένο κώδικα μεταξύ τους.

CREODIAS

Η υπηρεσία CREODIAS είναι μια από τις τέσσερις πλατφόρμες αποθήκευσης και επεξεργασίας γεωχωρικών δεδομένων στο πλαίσιο του Ευρωπαϊκού προγράμματος παρακολούθησης της γης Copernicus. Η πλατφόρμα παρέχει τα περισσότερα απ' τα δεδομένα καθώς και τις υπηρεσίες του προγράμματος Copernicus. Πιο συγκεκριμένα, η πλατφόρμα περιέχει περισσότερα από 18 Petabyte όγκου παρατηρησιακών δεδομένων τα οποία είναι άμεσα διαθέσιμα στους χρήστες. Οι πιο σημαντικές εφαρμογές που παρέχει η πλατφόρμα είναι οι εξής:

- (i) Το *Portal* το οποίο είναι το κέντρο για την διαχείριση και επεξεργασία του προσωπικού λογαριασμού του χρήστη. Μέσα απ' αυτό ο χρήστης έχει τη δυνατότητα να επικοινωνεί και να ενημερώνεται με άλλους χρήστες για διάφορες εφαρμογές που αφορούν την πλατφόρμα.
- (ii) Τον περιηγητή παρατηρησιακών δεδομένων (*EO Browser*) ο οποίος δίνει τη δυνατότητα στον χρήστη να οπτικοποιεί τα επιλεγμένα παρατηρησιακά δεδομένα.
- (iii) Τον *EO Finder* ο οποίος αποτελεί μια μηχανή αναζήτησης η οποία δίνει τη δυνατότητα στους χρήστες να αναζητούν δεδομένα θέτοντας συγκεκριμένους περιορισμούς, όπως ως προς τον χρόνο τον οποίο αποκτήθηκαν τα δεδομένα, τον τόπο αλλά και το είδος της συλλογής.
- (iv) Το *Cloud Dashboard* το οποίο αποτελεί ένα εργαλείο επισκόπησης και διαχείρισης των επεξεργαστηκών πόρων των χρηστών.

Όσον αφορά τους υπολογιστικούς πόρους, η πλατφόρμα προσφέρει μια σειρά από εικονικές μηχανικές με διαφορετικά λειτουργικά συστήματα, μεγάλους χώρους αποθήκευσης και ένα αρκετά γρήγορο δίκτυο διασύνδεσης. Τα είδη των συνόλων δεδομένων που προσφέρονται αφορούν τέσσερις κατηγορίες: 1) Την ατμόσφαιρα, 2) Καταστάσεις έκτακτης ανάγκης, 3) την Ξηρά και την 4) Θάλασσα. Τα δεδομένα είναι διαθέσιμα μέσω των δορυφόρων *Sentinel-1A & Sentinel-1B*, *Sentinel-2A & Sentinel-2B*, *Sentinel-3A & Sentinel-3B*, *Sentinel-5P*, *Landsat-5*, *Landsat-7*, *Landsat-8*, *Envisat*, *SMOS*. Στα αρνητικά της υπηρεσίας είναι ότι δεν διαθέτει κάποιο ενσωματωμένο σύστημα διεπαφής χρήστη με αποτέλεσμα να χρειάζεται αρκετός χρόνος απ' τη μεριά των χρηστών έτσι ώστε να εκμεταλλευτούν όλες τις δυνατότητες των υπολογιστικών πόρων που προσφέρει η πλατφόρμα.

WEkEO

Η υπηρεσία *WEkEO* είναι το σημείο αναφοράς του προγράμματος *Copernicus* για την αναζήτηση και επεξεργασία δεδομένων που αφορούν το περιβάλλον. Διαθέτει έναν διαδραστικό χάρτη μέσω του οποίου οι χρήστες μπορούν να αναζητήσουν τα δεδομένα ενδιαφέροντος και να τα οπτικοποιήσουν στον χάρτη με απλό και άμεσο τρόπο. Η υπηρεσία απευθύνεται σε μια μεγάλη πληθώρα χρηστών όπως σε ακαδημαϊκούς, επιστήμονες, επιχειρηματίες, εκπαιδευτικούς κ.τ.λ. Στα προτερήματα της πλατφόρμας είναι το γεγονός ότι αποτελεί τη μοναδική πλατφόρμα του προγράμματος *Copernicus* η οποία παρέχει μια ολοκληρωμένη υπηρεσία υποστήριξης των χρηστών της (*User Support Service*). Η υπηρεσία υποστήριξης στελεχώνεται από ειδικούς οι οποίοι μπορούν να φέρουν εις πέρας όλων των ειδών αιτημάτων σχετικά με προβλήματα επικοινωνίας, προβλήματα εγκατάστασης βιβλιοθηκών επεξεργασίας εικόνας ή και ακόμα να παράσχουν πληροφορίες σχετικά με θέματα δομών δεδομένων και εφαρμογών.

Για την εξοικείωση των νέων χρηστών με την πλατφόρμα, η υπηρεσία παρέχει εντελώς δωρεάν 12 *Webinars* και 4 *Training Sessions* μέσω του καναλιού της στο [YouTube](#). Η πρόσβαση σε όλα τα δεδομένα του προγράμματος *Copernicus*, στα *Jupyter Notebooks* και στις υπηρεσίες υποστήριξης χρηστών είναι εντελώς δωρεάν ενώ για πιο εξειδικευμένες παροχές όπως εικονικές μηχανές και πόρων δικτύου είναι επί πληρωμή.

Σε αντίθεση με τις υπόλοιπες υπηρεσίες του προγράμματος *Copernicus* η υπηρεσία *WEkEO* μέσω του *JupyterHub* παρέχει ένα ενσωματωμένο περιβάλλον στο οποίο οι χρήστες μπορούν να αναπτύσσουν κώδικα Python για την ανάλυση και επεξεργασία των δεδομένων ενδιαφέροντος που προσφέρει η υπηρεσία. Επιπροσθέτως, στο Hub υπάρχουν διαθέσιμα *Python Notebooks* τα οποία δίνουν πληροφορίες για το πώς γίνεται η λήψη δεδομένων από την πλατφόρμα *WEkEO* στο περιβάλλον του *Jupyter Notebook*. Άλλα notebook αντιστοιχούν σε training sessions σχετικά με εφαρμογές που σχετίζονται με την ατμόσφαιρα, την ξηρά και τους ωκεανούς.

Mundi

Η διαδικτυακή υπηρεσία *Mundi* είναι άλλη μια υπηρεσία του προγράμματος *Copernicus* η οποία παρέχει στους χρήστες της πρόσβαση για επεξεργασία και ανάλυση των δεδομένων του προγράμματος. Η πρόσβαση στα δεδομένα του προγράμματος είναι εντελώς δωρεάν μέσω της υπηρεσίας. Η πρόσβαση και η επεξεργασία των δεδομένων γίνεται μέσω του *Tenant*, του περιβάλλοντος που παρέχει η υπηρεσία. Σε αυτό το περιβάλλον, οι χρήστες μπορούν να αποθηκεύουν τα δεδομένα, να εκτελούν τους αλγορίθμους που έχουν υλοποιήσει και να αποθηκεύουν τα αποτελέσματα των αναλύσεων. Όπως και στις υπόλοιπες υπηρεσίες του Κοπέρνικου, η πρόσβαση σε πιο εξειδικευμένες υπηρεσίες, όπως σε εικονικές μηχανές και δίκτυο είναι επί πληρωμή.

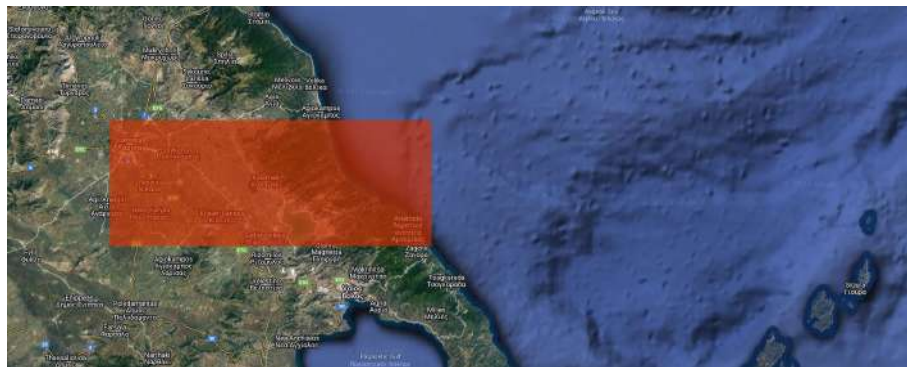
Η υπηρεσία παρέχει πρόσβαση σε όλα τα δεδομένα του προγράμματος *Copernicus*. Συνολικά τα δεδομένα που είναι αποθηκευμένα στην υπηρεσία φτάνουν τα 40 Petabyte. Ειδικότερα, στην πλατφόρμα εισέρχονται καθημερινά περίπου 17 Terabyte νέων δεδομένων που συλλέγονται από τους δορυφόρους. Ο όγκος αυτός αντιστοιχεί περίπου στο 95% των νέων δεδομένων που συλλέγονται από το πρόγραμμα *Copernicus*. Έτσι, η υπηρεσία παρέχει άμεσα στους χρήστες της τα πιο πρόσφατα δεδομένα. Η πρόσβαση στα δεδομένα γίνεται με τρεις διαφορετικούς τρόπους: 1) Μέσω ενός καταλόγου αναζήτησης, 2) Μέσω ενός διαδραστικού χάρτη και 3) μέσω του *Download Service*.

Η ανάγνωση των δεδομένων γίνεται μέσω του *Mundi Explore* το οποίο δίνει τη δυνατότητα οπτικοποίησης των δεδομένων, το μέγεθος των αρχείων αλλά και διάφορες στατιστικές που αφορούν τα δεδομένα ενδιαφέροντος. Επιπλέον, μια άλλη σημαντική υπηρεσία που προσφέρει η πλατφόρμα είναι το *Mundi Jupyter Notebook* το οποίο δίνει τη δυνατότητα ανάλυσης και επεξεργασίας των δεδομένων μέσω της γλώσσας προγραμματισμού Python. Όπως και στην περίπτωση του *WEkEO* μέσω της υπηρεσίας παρέχονται διάφορα notebooks τα οποία δίνουν τις βασικές οδηγίες και λειτουργίες της υπηρεσίας.

Ζητούμενο 2

Εκφώνηση Για την περιοχή μελέτης παρουσιάστε ένα φυσικό και ένα ψευδέχρωμο σύνθετο καθώς και το δείκτη NDVI για την ημερομηνία με τα λιγότερα σύννεφα κατά το 2019.

Σε αυτό το σημείο, όπως και στο πρώτο μέρος της τεχνικής έκθεσης απαντάμε στα ζητούμενα χρησιμοποιώντας το περιβάλλον του *GEE*. Στην παρακάτω εικόνα βλέπουμε την περιοχή μελέτης με την οποία θα ασχοληθούμε στα παρακάτω.



Σχήμα 16: Η περιοχή μελέτης.

Με τις παρακάτω εντολές επιλέγουμε τη συλλογή εικόνων του συνόλου δεδομένων [USGS Landsat 8 Collection 1 Tier 1 Raw Scenes](#) και κρατάμε μόνο τις εικόνες που περιέχουν το παραπάνω πολύγωνο. Επίσης, ορίζουμε τη συνάρτηση *addVariables* μέσω της οποίας υπολογίζουμε τους δείκτες NDVI, NDWI, EVI σε μια συγκεκριμένη εικόνα. Τέλος, εφαρμόζουμε αυτή τη συνάρτηση πάνω σε όλη τη συλλογή εικόνων που έχουμε κρατήσει.

```
//Filtering data

var l8 = L8.filter(ee.Filter.lt('CLOUD_COVER', 20)).
filterBounds(ROI);

// Adding Indices

var calculate_indices = function(image){

  // EVI = 2.5*(NIR - Red) / (NIR + 6*Red - 7.5*Blue + 1)
  var nir = image.select('B5');
  var red = image.select('B4');
  var blue = image.select('B2');
  var evi = (nir.multiply(2.5).subtract(red.multiply(2.5)).divide(nir.add(red.multiply(
    6)).
    add(blue.multiply(-7.5)).add(1))).rename('EVI');
```



```

var ndvi = image.normalizedDifference(['B5', 'B4']).float().rename('NDVI');
var ndwi = image.normalizedDifference(['B5', 'B6']).rename('NDWI');

return image.addBands(ndvi).addBands(ndwi).addBands(evi);
};

var l8 = l8.map(calculate_indices);

```

Με τις παρακάτω εντολές ορίζουμε τη χρωματική παλέτα που θα χρησιμοποιήσουμε για την αναπαράσταση της RGB εικόνας. Τέλος, οπτικοποιούμε την εικόνα για την περιοχή μελέτης για το φυσικό σύνθετο (RGB).

```

// Step 2 - Indices of image with less cloud cover

// RGB on ROI - 2019

var RGB_viz = {min: '8000', max: '10500',
bands: ['B4', 'B3', 'B2']}; // RGB visualization

var RGB_clearest_viz = {min: '6500', max: '9000',
bands: ['B4', 'B3', 'B2']}; // RGB clearest visualization

Map.addLayer(l8.filterDate('2019-01-01',
'2019-12-31').sort('CLOUD_COVER').median(), RGB_viz, 'RGB 2019');

Map.addLayer(l8.filterDate('2019-01-01',
'2019-12-31').sort('CLOUD_COVER').first().clip(ROI), RGB_clearest_viz, 'Clearest RGB
2019');

```

Παρακάτω βλέπουμε και την εικόνα που προκύπτει.



Σχήμα 17: Η εικόνα με τη μικρότερη νεφοκάλυψη σε φυσικό σύνθετο (RGB).

Κατά παρόμοιο τρόπο, ορίζουμε και τη χρωματική παλέτα για το ψευδέχρωμο σύνθετο και προσθέτουμε στα layer του χάρτη την εικόνα με τη μικρότερη νεφοκάλυψη και για αυτήν την αναπαράσταση της περιοχής μελέτης.

```

// False Color on ROI - 2019

var False_viz = {min: '6500', max: '15000',
bands: ['B5', 'B4', 'B3']}; // Pseudo visualization

Map.addLayer(l8.filterDate('2019-01-01',
'2019-12-31').sort('CLOUD_COVER').first().clip(ROI), False_viz, 'Clearest False color
2019');

```

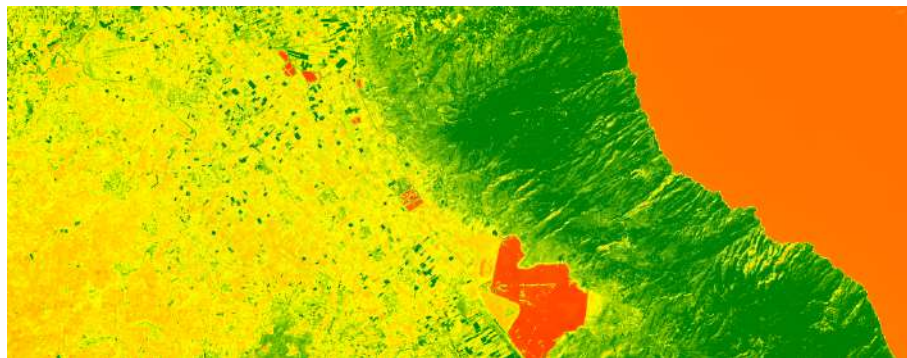
Παρακάτω βλέπουμε και το αποτέλεσμα που προκύπτει απ' τις παραπάνω εντολές.



Σχήμα 18: Η εικόνα με τη μικρότερη νεφοκάλυψη σε ψευδέχρωμο σύνθετο.

Τέλος, επαναλαμβάνουμε και την ίδια διαδικασία για την περίπτωση της εικόνας με τη μικρότερη νεφοκάλυψη χρησιμοποιώντας τον δείκτη βλάστησης *NDVI* αυτή τη φορά.

```
// NDVI clearest on ROI - 2019  
  
var NDVI_viz = {min: '-0.1', max: '0.35', bands: 'NDVI',  
palette: 'orangered, yellow, green'}; // NDVI visualization  
  
Map.addLayer(l8.filterDate('2019-01-01',  
'2019-12-31').sort('CLOUD_COVER').first().clip(ROI), NDVI_viz, 'Clearest NDVI 2019');
```



Σχήμα 19: Η εικόνα με τη μικρότερη νεφοκάλυψη μέσω του δείκτη *NDVI*.

Ζητούμενο 3

Εκφώνηση Παρουσιάστε και περιγράψτε το χάρτη με τις μέγιστες για το 2019 τιμές *NDVI* και τον χάρτη με τις αντίστοιχες ημερομηνίες (*DOY*) για τις οποίες είχαμε αυτές τις τιμές. Πραγματοποιήστε το ίδιο για το 2018 και συγκρίνετε τα αποτελέσματα. Που οφείλονται οι διαφορές;

Όπως και στην περίπτωση του 4ου Βήματος, σε αυτό το σημείο απεικονίζουμε την εικόνα που παράγεται κρατώντας τα εικονοστοιχεία της κάθε εικόνας όπου παρουσιάστηκε η μέγιστη τιμή του δείκτη *NDVI*. Η σύνθεση αυτής της εικόνας επιτυγχάνεται μέσω *qualityMosaic*. Επιπλέον, εφαρμόζουμε και τη συνάρτηση *addDOY* που χρησιμοποιήσαμε και στο βήμα 4 έτσι ώστε να καταγράψουμε την ημέρα του χρόνου που σημειώθηκε η εκάστοτε μέγιστη τιμή. Αυτά συνοψίζονται στις παρακάτω εντολές.

```
// Step 3
var NDVI_greenest_viz = {min: '-0.15', max: '0.55', bands: 'NDVI',
palette: 'orangered, yellow, green'};

var Season_viz = {min: '0', max: '366', bands: 'doy',
palette: 'lightblue, green, F37637, orangered'
};

var greenest_pixels2019 = l8.filterDate('2019-01-01', '2019-12-31')
.map(addDOY).qualityMosaic('NDVI').select('NDVI', 'doy');

// Quality Mosaic for 2019
Map.addLayer(greenest_pixels2019, NDVI_greenest_viz,
'Greenest pixels for year 2019');
```

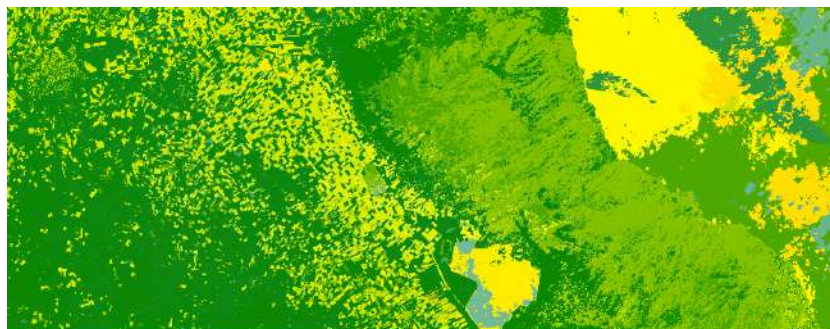
Στην παρακάτω εικόνα βλέπουμε και το αποτέλεσμα που παίρνουμε.



Σχήμα 20: Η εικόνα με τις μέγιστες τιμές του δείκτη *NDVI* για το 2019.

Στις παρακάτω εντολές ορίζουμε τη χρωματική παλέτα για την αναπαράσταση της εικόνας η οποία δείχνει τις ημέρες που σημειώθηκε η μέγιστη τιμή για το κάθε pixel. Για την απεικόνιση χρησιμοποιούμε τέσσερα χρώματα καθένα απ' τα οποία αντιστοιχεί σε μια εποχή του χρόνου (Χειμώνας, Άνοιξη, Καλοκαίρι, Φθινόπωρο). Με ανοικτό μπλέ αναπαριστούμε τις ημέρες που αντιστοιχούν στον Χειμώνα, με πράσινο στην Άνοιξη, με κίτρινο στο Καλοκαίρι και με πορτοκαλί-κόκκινο στο Φθινόπωρο.

```
var Season_viz = {min: '0', max: '366', bands: 'doy',
palette: 'lightblue, green, yellow, orangered'
};
// Seasonality Mapping for 2019
Map.addLayer(greenest_pixels2019, Season_viz,
'Seasonality Mapping 2019');
```

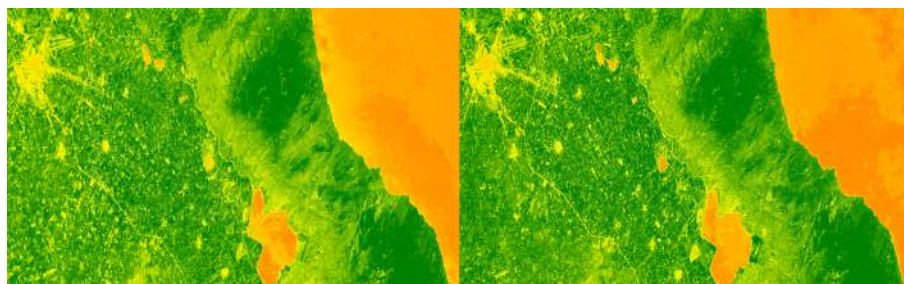


Σχήμα 21: Seasonality Mapping για το 2019.

Όπως βλέπουμε οι περισσότερες μέγιστες τιμές του *NDVI* εμφανίζονται κατά τη διάρκεια της Άνοιξης και του Καλοκαιριού. Φυσικά αυτό το αποτέλεσμα είναι αναμενόμενο αφού ο δείκτης *NDVI* αντιστοιχεί στον δείκτη βλάστησης και έτσι είναι λογικό να μεγιστοποιείται αυτές τις περιόδους. Στα παρακάτω συγκρίνουμε τις δύο παραπαπάνω εικόνες με τις αντίστοιχες εικόνες για το 2018.

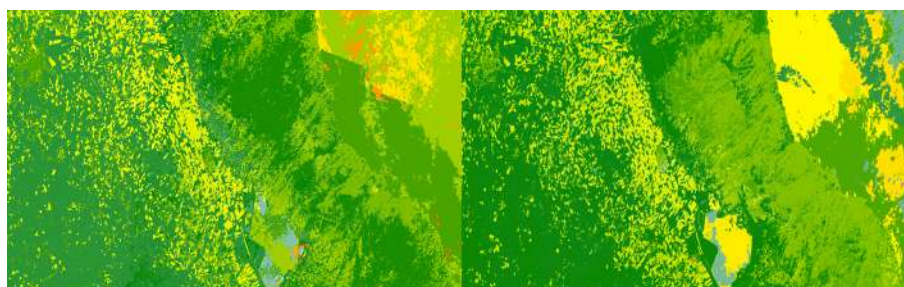
```
var greenest_pixels2018 = l8.filterDate('2018-01-01', '2018-12-31')
.map(addDOY).qualityMosaic('NDVI').select('NDVI', 'doy');

// Quality Mosaic for 2018
Map.addLayer(greenest_pixels2018.clip(ROI), NDVI_greenest_viz,
'Greenest pixels for year 2018');
// Seasonality Mapping for 2018
Map.addLayer(greenest_pixels2018.clip(ROI), Season_viz,
'Seasonality Mapping 2018');
```



Σχήμα 22: Μέγιστες τιμές *NDVI* για το 2018 (αριστερά) και για το 2019 (δεξιά).

Όπως βλέπουμε απ' τις δύο παραπάνω εικόνες δεν υπάρχει μεγάλη διαφορά μεταξύ τους. Αυτό που παρατηρούμε είναι μια αμυδρή αύξηση της μέγιστης του δείκτη *NDVI* κατά το 2019 σε σχέση με το 2018 κυρίως στην περιοχή που αντιστοιχεί στις αγροτικές καλλιέργειες της περιοχής (κάτω αριστερά στις εικόνες). Παρακάτω βλέπουμε και τους χάρτες που απεικονίζουν τη χρονολογική άφιξη αυτών των μέγιστων τιμών για το 2018 και το 2019, αντίστοιχα.



Σχήμα 23: Seasonality Mapping για το 2018 (αριστερά) και για το 2019 (δεξιά).

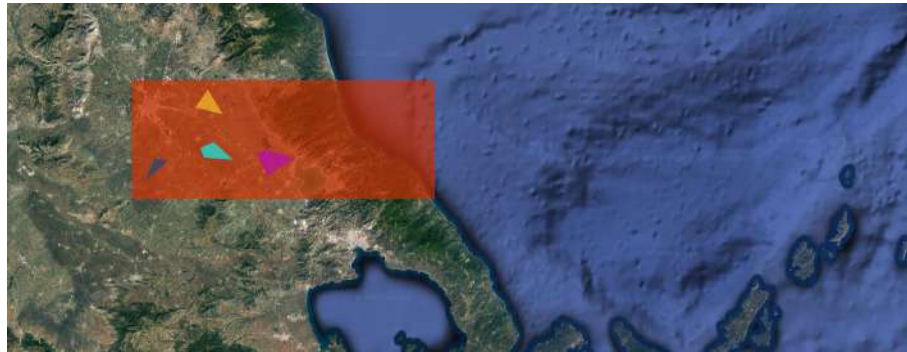
Και σε αυτή την περίπτωση παρατηρούμε παρόμοια συμπεριφορά και στις δύο περιπτώσεις. Δηλαδή, οι η άφιξη των μέγιστων τιμών για τον δείκτη βλάστησης κυρίως μεταξύ της περιόδου της Άνοιξης και του Καλοκαιριού.

Ζητούμενο 4

Εκφώνηση Για 4 διαφορετικά πολύγωνα στην περιοχή μελέτης, για όλα τα διαθέσιμα δεδομένα L8 και τις χρονοσειρές *NDVI* (ή άλλων δεικτών για τον αστικό χώρο, τα ύδατα, το έδαφος), περιγράψτε και

παρουσιάστε τις βέλτιστες αρμονικές καμπύλες που προσαρμόσατε δημιουργώντας συνθετικές χρονοσειρές.

Σε αυτό το σημείο δημιουργούμε συνθετικές χρονοσειρές των δεδομένων εφαρμόζοντας την τεχνική της αρμονικής παλινδρόμησης που χρησιμοποιήσαμε και στο βήμα 5 του πρώτου μέρους. Για να προσαρμόσουμε την καμπύλη για τα 4 διαφορετικά πολύγωνα κατασκευάζουμε μια συνάρτηση *print_harmonic_model* η οποία δέχεται ως όρισμα το επιλεγμένο πολύγωνο που θέλουμε να προσαρμόσουμε την καμπύλη και εξάγει το γράφημα της χρονοσειράς του μέσου δείκτη *NDVI* για όλα τα διαθέσιμα δεδομένα και την προσαρμοσμένη αρμονική καμπύλη που προκύπτει. Στα επόμενα βλέπουμε το σώμα της συνάρτησης και τα τέσσερα επιλεγμένα υπό-πολύγωνα που αντιστοιχούν σε αγροτικές εκτάσεις της περιοχής μελέτης.



Σχήμα 24: Τα τέσσερα πολύγωνα αγροτικών εκτάσεων στην περιοχή μελέτης.

```
var all_data_ndvi = 18.map(addVariables);
// A function to generate harmonic plot
var print_harmonic_model = function(data, poly, title){
// Harmonic trend -----
// Use these independent variables in the harmonic regression.
  var harmonicIndependents = ee.List(['constant', 't', 'cos', 'sin']);
// Add harmonic terms as new image bands.
  var harmonicLandsat = data.map(function(image) {
    var timeRadians = image.select('t').multiply(2 * Math.PI);
    return image
      .addBands(timeRadians.cos().rename('cos'))
      .addBands(timeRadians.sin().rename('sin'));
  });

// The output of the regression reduction is a 4x1 array image.
  var harmonicTrend = harmonicLandsat
    .select(harmonicIndependents.add('NDVI'))
    .reduce(ee.Reducer.linearRegression(harmonicIndependents.length(), 1));

// Turn the array image into a multi-band image of coefficients.
  var harmonicTrendCoefficients = harmonicTrend.select('coefficients')
    .arrayProject([0])
    .arrayFlatten([harmonicIndependents]);

// Compute fitted values.
  var fittedHarmonic = harmonicLandsat.map(function(image) {
    return image.addBands(
      image.select(harmonicIndependents)
        .multiply(harmonicTrendCoefficients)
        .reduce('sum')
        .rename('fitted'));
  });
```

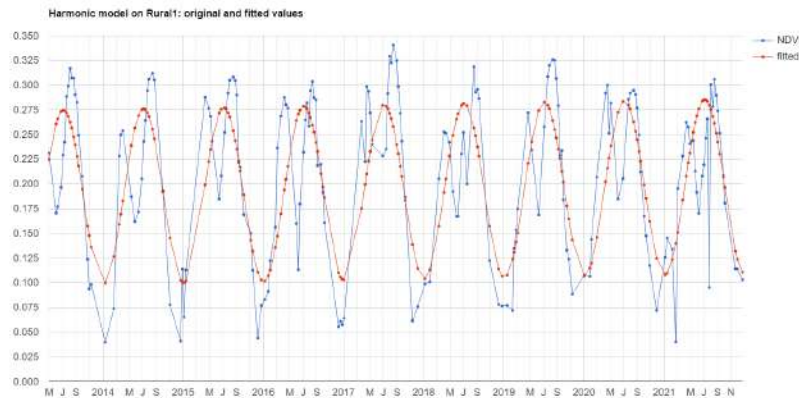
```

});

// Plot the fitted model and the original data at the ROI.
print(ui.Chart.image.series(
  fittedHarmonic.select(['fitted', 'NDVI']), poly, ee.Reducer.mean(), 30)
  .setSeriesNames(['NDVI', 'fitted'])
  .setOptions({
    title: 'Harmonic model' + title + ': original and fitted values',
    lineWidth: 1,
    pointSize: 3,
  }));
});
// ROI 1
print_harmonic_model(all_data_ndvi, Rural1, ' on Rural1');
// ROI 2
print_harmonic_model(all_data_ndvi, Rural2, ' on Rural2');
// ROI 3
print_harmonic_model(all_data_ndvi, Rural3, ' on Rural3');
// ROI 4
print_harmonic_model(all_data_ndvi, Rural4, ' on Rural4');

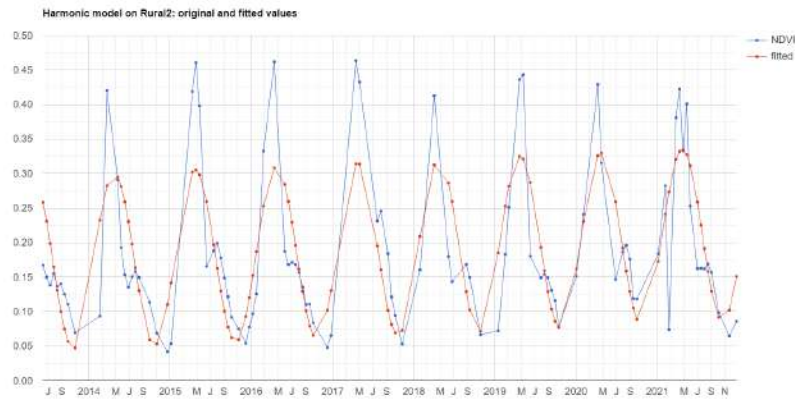
```

Στην επόμενη εικόνα βλέπουμε το γράφημα που αντιστοιχεί στο μοβ πολύγωνο του σχήματος 24.



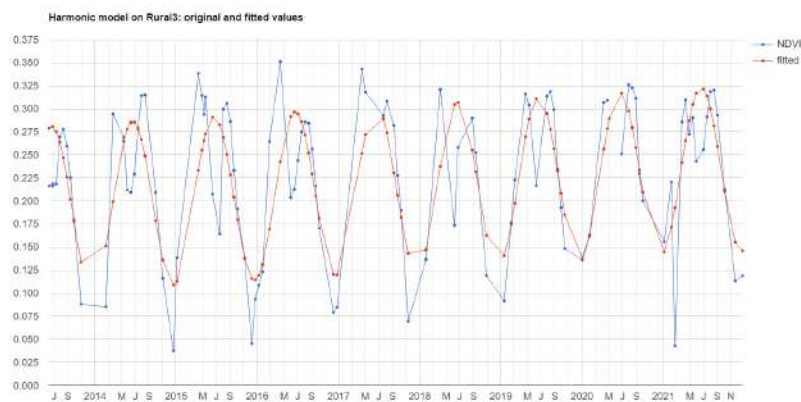
Σχήμα 25: Η αρμονική χρονοσειρά του μοβ πολυγώνου.

Όπως φαίνεται απ' το παραπάνω σχήμα ο μέσος όρος του δείκτη βλάστησης στο πολύγωνο είναι περιοδικός ως προς το χρόνο. Είναι εμφανές ότι μεταξύ των μηνών Μαΐου - Αυγούστου έχουμε τις υψηλότερες τιμές ενώ μεταξύ των μηνών Δεκεμβρίου - Ιανουαρίου έχουμε τις χαμηλότερες τιμές. Το ότι οι υψηλές τιμές του δείκτη συνεχίζονται και μέχρι τον Αύγουστο οφείλεται στο γεγονός ότι τα πολύγωνα που έχουμε επιλέξει αντιστοιχούν σε αγροτικές εκτάσεις και έτσι υπάρχει συνεχής άρδευση αυτών των εκτάσεων καθόλη τη διάρκεια αυτών των μηνών. Όπως βλέπουμε η υψηλότερη τιμή σημειώνεται στις 15 Αυγούστου του 2017 με τιμή 0.341 ενώ η χαμηλότερη τιμή σημειώνεται στις 22 Φεβρουαρίου του 2022 και στις 11 Ιανουαρίου του 2014 με τιμή 0.04. Στο επόμενο σχήμα βλέπουμε και το αντίστοιχο γράφημα που αντιστοιχεί στο μπλε πολύγωνο του σχήματος 24.



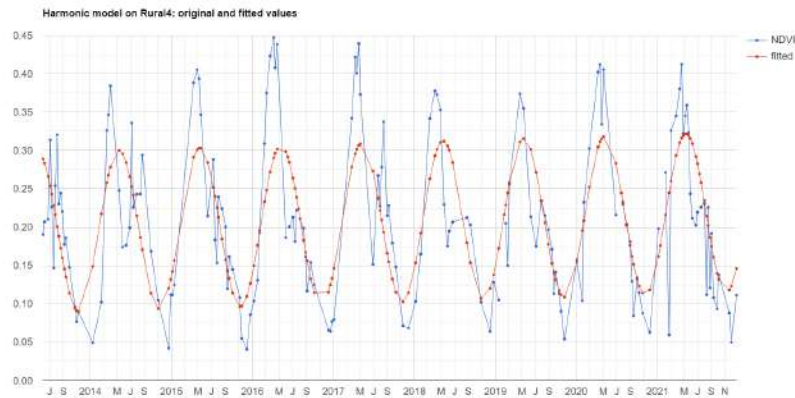
Σχήμα 26: Η αρμονική χρονοσειρά του μπλε πολυγώνου.

Όπως βλέπουμε το μπλε πολύγωνο εμφανίζει παρόμοια μορφή περιοδικότητας με το μοβ πολύγωνο. Οι μέγιστες τιμές του μέσου δείκτη *NDVI* σημειώνονται στις 27 Απριλίου του 2015 με 0.461, στις 13 Απριλίου του 2016 με 0.462 και στις 16 Απριλίου του 2017 με 0.463. Οι ελάχιστες τιμές σημειώνονται στις 20 Δεκεμβρίου του 2014 με 0.042 και στις 9 Δεκεμβρίου του 2016 με 0.048. Αυτό που παρατηρούμε ότι αυτό το πολύγωνο εμφανίζει μεγαλύτερες τιμές απ' το προηγούμενο αλλά και πιο απότομες μεταβολές. Παρακάτω βλέπουμε το αντίστοιχο γράφημα για το κίτρινο πολύγωνο του σχήματος 24.



Σχήμα 27: Η αρμονική χρονοσειρά του γαλάζιου πολυγώνου.

Και σε αυτό το πολύγωνο οι μέγιστες σημειώνονται μεταξύ του 2015 και του 2017 με τιμές μεταξύ 0.338 και 0.351. Όπως και στην περίπτωση των δύο προηγούμενων γραφημάτων μετά απ' το 2017 παρουσιάζεται ελαφριά πτώση του δείκτη φτάνοντας το 0.32 τον Αύγουστο του 2021. Τέλος, παρουσιάζουμε και το γράφημα του γαλάζιου πολυγώνου του σχήματος 24.

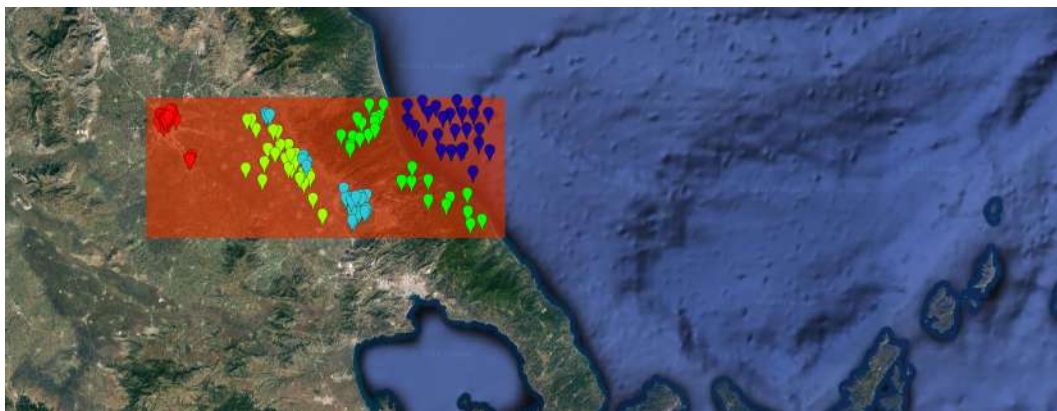


Σχήμα 28: Η αρμονική χρονοσειρά του γαλάζιου πολυγώνου.

Ζητούμενο 5

Εκφώνηση Για την ίδια ή διαφορετική περιοχή μελέτης, ψηφιοποιήστε διάφορες χρήσεις γης και πραγματοποιείτε ταξινόμηση στην διαθέσιμη χρονοσειρά. Χρησιμοποιείτε δυο διαφορετικούς ταξινομητές. Περιγράψτε τυχόν διαφορές που προκύπτουν στα αποτελέσματα της ταξινόμησης σε σχέση με τις διαθέσιμες ψηφιοποιημένες περιοχές ή τους διαφορετικούς ταξινομητές.

Για την ταξινόμηση διακρίνουμε πέντε διαφορετικές κατηγορίες στην περιοχή μελέτης: 1) Τον αστικό ιστό (κόκκινη απόχρωση), 2) τις αγροτικές εκτάσεις (ανοικτό πράσινο), 3) τις δασικές εκτάσεις (σκούρο πράσινο), 4) τις λίμνες (γαλάζιο) και 5) τις θαλάσσιες περιοχές (σκούρο μπλε). Για κάθε μια απ' τις παραπάνω κατηγορίες συλλέγουμε 25 δείγματα τα οποία θα αποτελέσουν το σύνολο εκπαίδευσης των μοντέλων που θα χρησιμοποιήσουμε. Για την ταξινόμηση θα χρησιμοποιήσουμε έναν ταξινομητή CART και ένα γραμμικό SVM. Στην παρακάτω εικόνα βλέπουμε την εικόνα που περιέχει το πολύγωνο της περιοχής μελέτης καθώς και τα δειγματοστοιχεία που επιλέγουμε για τις παραπάνω πέντε κατηγορίες.



Σχήμα 29: Το σύνολο εκπαίδευσης στην περιοχή μελέτης.

Επίσης, για κάθε μια απ' τις πέντε κατηγορίες επιλέγουμε και 50 σημεία τα οποία θα αποτελέσουν το σύνολο ελέγχου (test set) για την ταξινόμηση. Με τις επόμενες εντολές, αρχικοποιούμε τους δύο ταξινομητές, πραγματοποιούμε την εκτίμηση των παραμέτρων χρησιμοποιώντας το σύνολο εκπαίδευσης και ελέγχουμε τα αποτελέσματα μέσω του συνόλου ελέγχου.

```
// Step 5 - Classification Part

var classification_img = L8.filter(ee.Filter.lt('CLOUD_COVER', 20)).
filterBounds(ROI).mean().clip(ROI);

Map.addLayer(classification_img, RGB_viz, 'Classification RGB');

var features = classification_img.bandNames();
var training_pts = forest.merge(rural).merge(urban).merge(sea).merge(lake);
var test_pts = test_forest.merge(test_rural).merge(test_urban).merge(test_sea).
merge(test_lake);

var training_data = classification_img.sampleRegions({collection: training_pts,
  properties: ['class'], scale: 30
});

var test_data = classification_img.sampleRegions({collection: test_pts,
  properties: ['class'], scale: 30
});

var CART = ee.Classifier.smileCart();
// Create an SVM classifier with custom parameters.
var SVM = ee.Classifier.libsvm({
  kernelType: 'LINEAR',
  cost: 1
});

var CART = CART.train({features: training_data,
  classProperty: 'class', inputProperties: features});
var SVM = SVM.train({features: training_data,
  classProperty: 'class', inputProperties: features});
var CART_img = classification_img.classify(CART);
var SVM_img = classification_img.classify(SVM);

var Class_viz = {min: 0, max: 4,
palette: ['green', 'lightgreen', 'red', 'blue', '#689EDF']};

Map.addLayer(CART_img, Class_viz, 'Classified Image - CART');
Map.addLayer(SVM_img, Class_viz, 'Classified Image - SVM');

var evaluation = function(classifier, test_data, features, title){

  var confusionMatrix = ee.ConfusionMatrix(test_data.classify(classifier).errorMatrix({
    actual: 'class',
    predicted: 'classification'}));

  print(title);
  print('Confusion Matrix', confusionMatrix);
  print('Overall accuracy', confusionMatrix.accuracy());
};

evaluation(CART, test_data, features, 'CART Evaluation');
evaluation(SVM, test_data, features, 'SVM Evaluation');
```

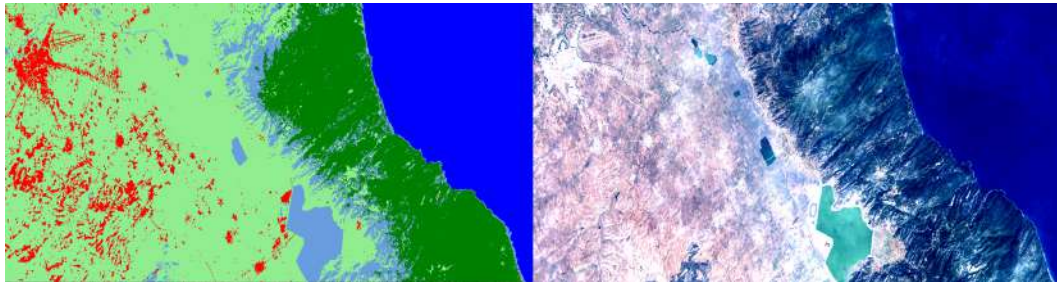
Η αξιολόγηση των μοντέλων γίνεται μέσω μιας συνάρτησης *evaluation* η οποία δέχεται ως ορίσματα τον ταξινομητή (*classifier*), το σύνολο ελέγχου (*test_data*), τα χαρακτηριστικά (*features*) και μια συμβολοσειρά (*title*) η οποία αντιστοιχεί στον τίτλο που θα εκτυπωθεί στην οθόνη ώστε να μπορούμε να διακρίνουμε σε ποιον ταξινομητή αναφερόμαστε κάθε φορά. Η συνάρτηση εκτυπώνει στην ο-

θόνη έναν πίνακα σύγχυσης (*confusion matrix*) αλλά και το ποσοστό ακρίβειας (*accuracy score*) που προκύπτει από το σύνολο ελέγχου. Στις επόμενες εικόνες βλέπουμε τις απεικονίσεις που παίρνουμε στον χάρτη σε σύγκριση με την πραγματική εικόνα για κάθε μια απ' τις δυο ταξινομήσεις χρησιμοποιώντας τα χρώματα που αναφέραμε προηγουμένως. Η πρώτη εικόνα αντιστοιχεί στην ταξινόμηση που προκύπτει μέσω του CART.



Σχήμα 30: Το αποτέλεσμα της ταξινόμησης μέσω του CART.

Στην παρακάτω εικόνα βλέπουμε και το αποτέλεσμα που προκύπτει μέσω του SVM.



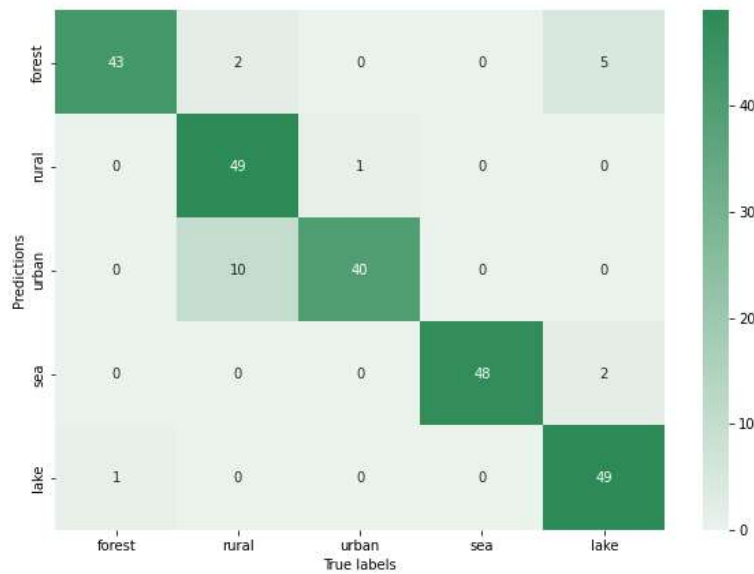
Σχήμα 31: Το αποτέλεσμα της ταξινόμησης μέσω του SVM.

Στον παρακάτω πίνακα βλέπουμε και την απόδοση των δύο μοντέλων με βάση το *accuracy score*.

Πίνακας 1: Η απόδοση των ταξινομητών

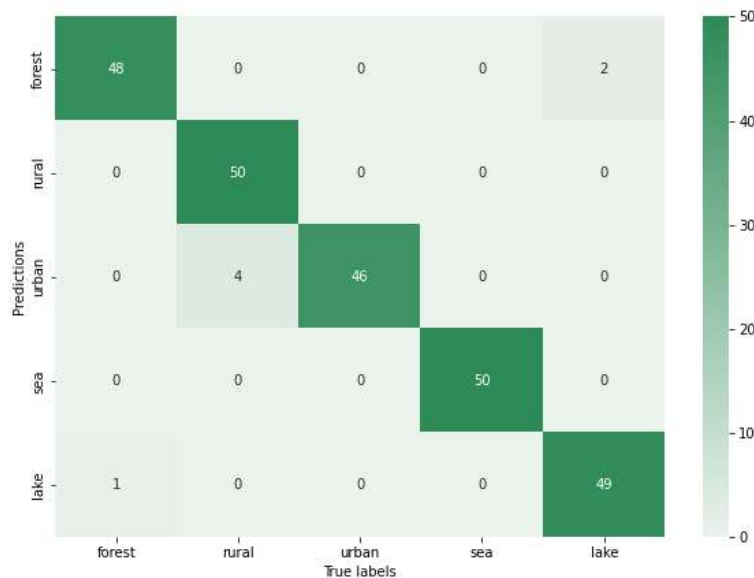
Ταξινομητής	Ακρίβεια
CART	0.916
SVM	0.972

Όπως βλέπουμε η επίδοση του SVM ήταν καλύτερη απ' αυτή του CART κατά 5% περίπου. Παρατηρώντας τις εικόνες στα σχήματα 30 και 31 βλέπουμε ότι ο ταξινομητής CART μπερδεύει σε μεγαλύτερο ποσοστό τον αστικό ιστό απ' ό,τι ο ταξινομητής SVM. Αυτό φαίνεται και καλύτερα στους πίνακες σύγχυσης που ακολουθούν, μέσα απ' τους οποίους μπορούμε να δούμε που ακριβώς κάνανε λάθος οι ταξινομητές.



Σχήμα 32: Ο πίνακας σύγχυσης του ταξινομητή CART.

Απ' τον παραπάνω πίνακα γίνεται εμφανές το πρόβλημα που παρατηρήσαμε και απ' τις εικόνες του σχήματος 30. Δηλαδή, ο ταξινομητής CART κατηγοριοποίησε 10 δείγματα ως αστικό ιστό ενώ στην πραγματικότητα αντιστοιχούν σε αγροτικές εκτάσεις. Παρακάτω βλέπουμε και τον πίνακα σύγχυσης που αντιστοιχεί στον ταξινομητή SVM.



Σχήμα 33: Ο πίνακας σύγχυσης του ταξινομητή SVM.

Ζητούμενο 6

Εκφώνηση Με βάση τα παραπάνω βήματα, υλοποιήστε ένα δικό σας *app* μέσω της GEE.

Σε αυτό το σημείο με βάση τα βήματα του πρώτου μέρους κατασκευάζουμε μια εφαρμογή στο περιβάλλον του GEE μέσω της οποίας ο χρήστης μπορεί να οπτικοποιεί για διάφορες περιοχές (πολύγωνα) της επιλογής του την εικόνα που προκύπτει υπολογίζοντας τη διάμεσο του κάθε *pixel* από όλες τις εικόνες που υπάρχουν διαθέσιμες για το έτος του 2019 μέσω του δορυφόρου *Landsat*. Επίσης, ο χρήστης μπορεί να οπτικοποιεί και τις αντίστοιχες εικόνες για τους δείκτες *NDVI*, *NDWI* καθώς και την εικόνα με τις μεγαλύτερες τιμές του δείκτη *NDVI* για όλο το 2019. Ένα άλλο χαρακτηριστικό της εφαρμογής είναι ότι δίνει τη δυνατότητα στον χρήστη να αφαιρεί τα πολύγωνα που έχει ορίσει και να δημιουργεί νέα πολύγωνα σε διαφορετικές περιοχές. Η ίδια δυνατότητα δίνεται στον χρήστη και για τα στρώματα (*layers*) που έχει προσθέσει στον χάρτη. Η πρόσβαση στην εφαρμογή γίνεται μέσω του συνδέσμου <https://chrisnick92.users.earthengine.app/view/app1>. Παρακάτω δίνουμε εν συντομία τις οδηγίες για τη χρήση της εφαρμογής.

- (i) Αρχικά ο χρήστης επιλέγει είτε χρησιμοποιώντας τον διαδραστικό χάρτη είτε τη μπάρα αναζήτησης για να εστιάσει στην περιοχή ενδιαφέροντος.
- (ii) Εν συνεχεία, μέσω της ένδειξης *rectangle* ή *polygon* ο χρήστης επιλέγει το είδος της γεωμετρίας που θέλει να σχεδιάσει στον χάρτη.
- (iii) Έχοντας σχεδιάσει το πολύγωνο στην περιοχή ενδιαφέροντος ο χρήστης επιλέγει το είδος του στρώματος που θέλει να οπτικοποιήσει στον χάρτη.
- (iv) Για να φανεί το αποτέλεσμα του επιλεγμένου στρώματος στον χάρτη ο χρήστης πρέπει πρώτα να αφαιρέσει το πλαίσιο της γεωμετρίας που έχει αρχικώς επιλέξει έτσι ώστε να μείνει μόνο η απεικόνιση του στρώματος στην οθόνη. Αυτό επιτυγχάνεται απλά πατώντας το κουμπί που αντιστοιχεί στο είδος της γεωμετρίας που επιλέχθηκε στο βήμα (ii).
- (v) Μέσω της επιλογής *Erase Layer* δίνεται η δυνατότητα στον χρήστη να αφαιρέσει από τον χάρτη το τελευταίο στρώμα που προστέθηκε κατά σειρά. Αντίστοιχη λειτουργία έχει και η επιλογή *Erase Geometry*.

Αναφορές - Links

- [1] European Space Agency. *Vegetation Indices*. https://www.esa.int/SPECIALS/Eduspace_Global_GR/SEMRTITWT1H_0.html.
- [2] Copernicus. *DIAS*. Available Online: <https://www.copernicus.eu/en/access-data/dias>.
- [3] CREODIAS. *What is CREODIAS?* Available online: <https://creodias.eu>.
- [4] Vitor CF Gomes, Gilberto R Queiroz, and Karine R Ferreira. “An overview of platforms for big earth observation data management and analysis”. In: *Remote Sensing* 12.8 (2020), p. 1253.
- [5] Mundi. Available online: <https://mundiwebservices.com>.
- [6] ONDA. Available online: <https://www.onda-dias.eu/cms/>.
- [7] Amazon Web Services. *Open Data on AWS*. 2020. Available online: <https://aws.amazon.com/opendata/?wwps-cards.sort-by=item.additionalFields.sortDate&wwps-cards.sort-order=desc>.
- [8] sobloo. Available online: <https://sobloo.eu>.
- [9] WEkEO. Available online: <https://www.wekeo.eu>.