salsa
small farms
small food businesses and
sustainable food security

30.04.2019
Deliverable 2.3

Methodological report and guidance for the use of SENTINEL-2 data in assessing and monitoring of small farms and crop production.

WP2 Estimation of the distribution and production capacity of small farms

Prepared under contract from the European Commission
Project number: 677363
Collaborative project
Horizon2020

Project acronym	**SALSA**
Project full title	Small Farms, Small Food Businesses and Sustainable Food Security
Duration	April 2016 – March 2020
Coordinating organisation	Universidade de Évora (UEvora)
Project coordinator	Teresa Pinto-Correia
Project website:	www.salsa.uevora.pt
Deliverable title	Methodological report and guidance for the use of SENTINEL-2 data in assessing and monitoring of small farms and crop production.
Deliverable number	2.3
Work package	WP2
Authors	Sérgio Godinho; Nuno Guiomar; Rui Machado; Teresa Pinto Correia; Theodore Tsiligiridis; Katerina Ainali

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

Contents

Version	*Date*	*Change*
1.0	23.03.2018	Deliverable 2.3 (draft version)
2.0	30.04.2019	Deliverable 2.3 (Final version)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

1. Introduction

1.1. Outline of the Deliverable D.2.3

This deliverable is a report on the main methodological steps implemented in the framework of the Small Farms, Small Food Business and Sustainable Food Security (SALSA) project in task 2.3 of Work Package 2 (WP2) to produce the Output 3, which is a crop type map in small farms context in each reference region (see D.2.4 report pag. 5). Accordingly, this report is not focused on presenting the results about the usefulness of Sentinel data in providing accurate crop type maps, but rather in providing a detailed explanation about the main methodological chain applied for the use of Sentinel-1A/B and Sentinel-2A/B data for crop type mapping in small farms. The results about crop types and production potential, as well as the assessment of the effectiveness of Sentinel data are presented in Deliverable 2.4. This methodological guidance can be seen as a good-practice 'toolkit' which describes the main technical and methodological steps to better use spatial and statistical tools (e.g. ArcMap and R), as well as Sentinel data, to create crop type maps, including the strategy to select the field points, which is a crucial step in the whole process. It is as such that this methodological guidance plays a key role in providing useful recommendations for those dealing with remote sensing data for crop type mapping.

2. Workflow to produce a crop type map using Sentinel data (output 3)

Three main steps were implemented to produce a crop type map for each reference region:

i). collection of reference crop data in each region;
ii). quality control of collected reference crop data; and
iii). image processing and classification.

i). collection of reference crop data in each region

This step involved intense fieldwork for crop type data collection to be used as calibration/validation datasets in the image classification procedure. Due to the very high costs of field campaigns to collect enough calibration/validation data, a methodological approach was developed and implemented to select a minimum number of points to be collected in each region, so that the effort dedicated to this task was contained.

To balance between budget, time and accuracy, we decided as the baseline to produce the reference crop dataset the following:

- select an average of 500 points in each reference region;
- select 25 squares (2x2km) per reference region;
- select 20 points in each square;

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

- stratify the number of points using the area of each class obtained from the unsupervised classification. The class with the largest area should receive a maximum of 50 points; minimum points per class are 10.

To ensure a robust, balanced, and heterogeneously distributed sample in each reference region that encompasses agricultural cover diversity and, simultaneously, minimizes field work efforts by spatially constraining the sample points to the selected squares (e.g. Song et al., 2017), we have applied a six-stage methodology to define the spatial distribution of the sample points. Basically, the approach aims to combine two main criteria to select the sample points; i) the agricultural landscape diversity, and ii) the accessibility of each square. The idea is to select the squares with the highest agricultural diversity and relatively high road density to facilitate the field work.

Stage 1, 2 and 3 is dedicated to quantify agricultural diversity, stage 4 to determine the degree of accessibility in each 2x2km squares, stage 5 is dedicated to the squares selection by combining results from stage 1 to 4, and finally, stage 6 is devoted to the field points selection.

1. Create fishnet with 2 x2 km size using the ArcGis tool “Create Fishnet” in Data Management Tools, Feature Class;
    a. Clip the created fishnet using the reference region limits;
    b. Delete all the border squares;
2. Unsupervised Random Forest classification
    a. Download Sentinel-2 TCI band (true colour composition). Here we only used these bands to speed up the process;
    b. Clip the TCI band using the “agriculture mask” shapefile (Delivreable 2.2);
    c. Run the unsupervised classification using the R script developed for this purpose (ANEXX I); the result will be a tif file with the number of classes/clusters obtained by the model;
    d. convert the unsupervised classification from raster to shapefile;
    e. calculate the area for each polygon, and after delete all the polygons <0.03ha;
    f. compute the number of points per each class;

For the unsupervised random forest classification the *Gap* statistics from the *clusGap* R function was used to determine the optimal number of clusters (classes).

3. Compute Shannon Evenness Index (SEI) to assess agricultural landscape heterogeneity

After unsupervised classifications the overall agricultural land cover diversity along a regular grid of 2x2 km in each reference region was assessed by computing the Shannon Evenness Index (SEI):

$$\mathrm{SEI} = \frac{\mathrm{SDI}}{\ln \mathrm{n}}$$

where *n* is the number of land cover types (clusters from point 2) and SDI is Shannon Diversity Index which, in turn, was calculated as follows:

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 677363

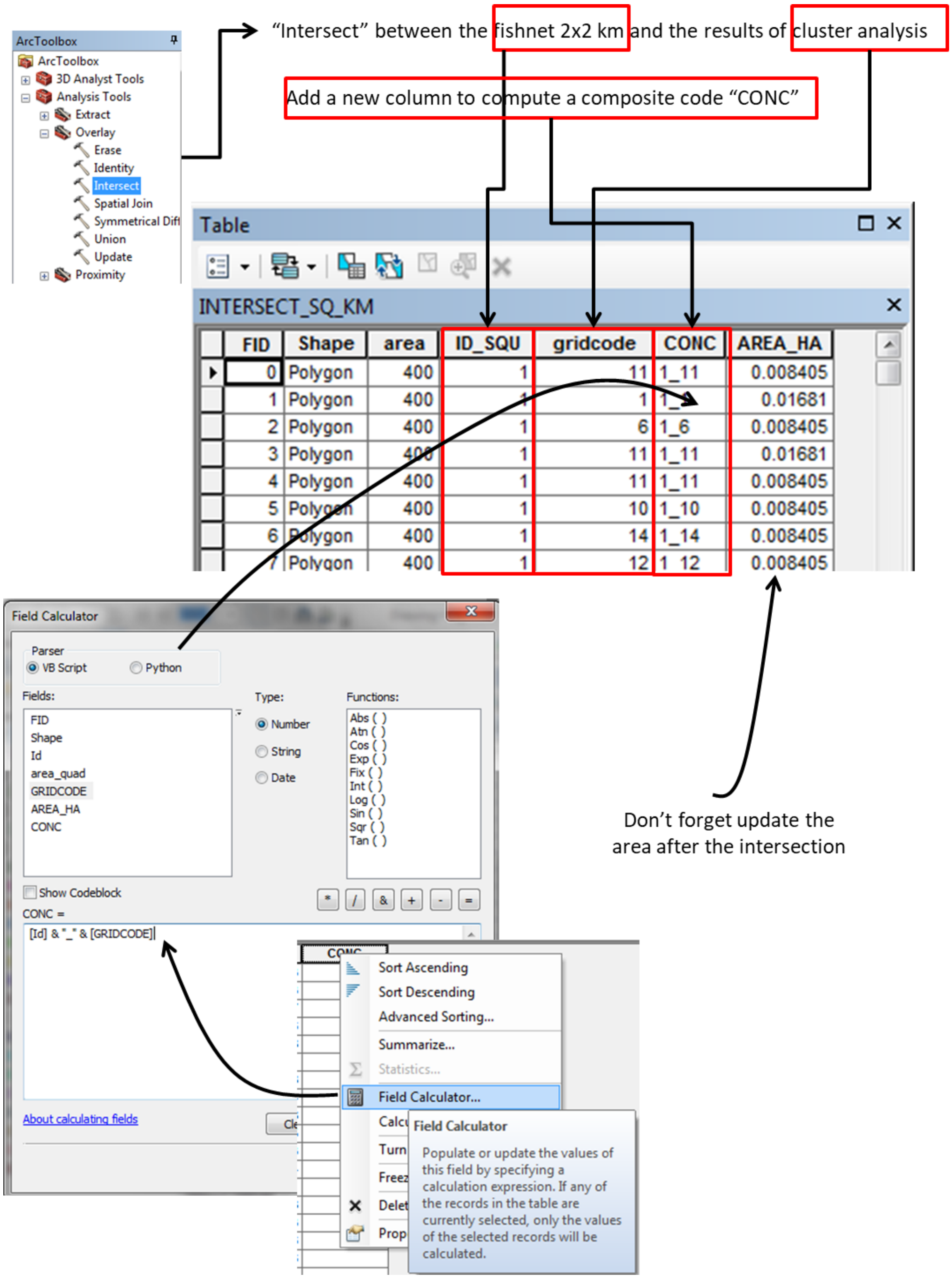
$$\mathrm{SDI} = -\sum_{\mathrm{i}=1}^{\mathrm{n}} (\mathrm{p_i} \times \ln \mathrm{p_i})$$

where p_i is the proportional abundance of the ith type. Here pi represents the proportion of each cluster type, and *n* the total number of cluster types in a reference region.

Bellow a set of figures summarizes how to compute Shannon Diversity Index (SDI) and Shannon Evenness Index (SEI) in ArcGIS using vector data.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

a) Intersection between the fishnet 2x2 km and the results of cluster analysis



INTERSECT_SQ_KM

FID	Shape	area	ID_SQU	gridcode	CONC	AREA_HA
0	Polygon	400	1	11	1_11	0.008405
1	Polygon	400	1	1	1_	0.01681
2	Polygon	400	1	6	1_6	0.008405
3	Polygon	400	1	11	1_11	0.01681
4	Polygon	400	1	11	1_11	0.008405
5	Polygon	400	1	10	1_10	0.008405
6	Polygon	400	1	14	1_14	0.008405
7	Polygon	400	1	12	1_12	0.008405

Parser: [✓ VB Script] Python

Type: [✓ Number] String Date

CONC =

```
[Id] & "_" & [GRIDCODE]
```

Figure 1 – Intersection between the fishnet 2x2 km

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 677363

b) Build a summary table based on the concatenated code (CONC from Figure 1)

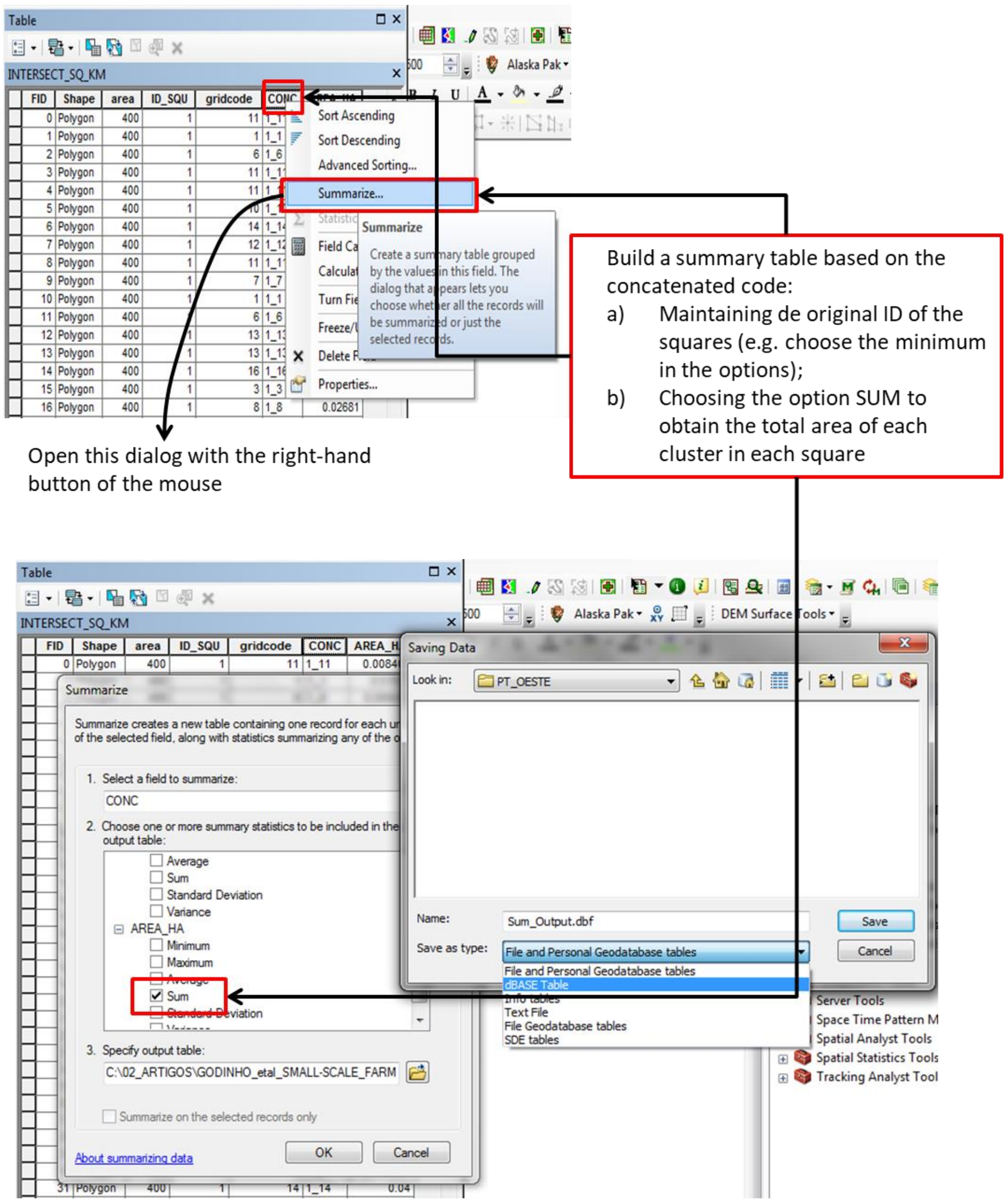


Table
INTERSECT_SQ_KM
Sort Ascending
Sort Descending
Advanced Sorting...
Summarize...
Summarize
Create a summary table grouped by the values in this field. The dialog that appears lets you choose whether all the records will be summarized or just the selected records.
Properties...
Saving Data
Look in: PT_OESTE
Save
Cancel
File and Personal Geodatabase tables
dBASE Table
Text File
File Geodatabase tables
SDE tables

FID	Shape	area	ID_SQU	gridcode	CONC
0	Polygon	400	1	11	1_1
1	Polygon	400	1	1	1_1
2	Polygon	400	1	6	1_6
3	Polygon	400	1	11	1_11
4	Polygon	400	1	11	1_1
5	Polygon	400	1	10	1_
6	Polygon	400	1	14	1_14
7	Polygon	400	1	12	1_12
8	Polygon	400	1	11	1_11
9	Polygon	400	1	7	1_7
10	Polygon	400	1	1	1_1
11	Polygon	400	1	6	1_6
12	Polygon	400	1	13	1_13
13	Polygon	400	1	13	1_13
14	Polygon	400	1	16	1_16
15	Polygon	400	1	3	1_3
16	Polygon	400	1	8	1_8

Open this dialog with the right-hand button of the mouse

Build a summary table based on the concatenated code:

a) Maintaining de original ID of the squares (e.g. choose the minimum in the options);

b) Choosing the option SUM to obtain the total area of each cluster in each square

Summarize

Summarize creates a new table containing one record for each u[...] of the selected field, along with statistics summarizing any of the o[...]

1. Select a field to summarize: CONC

2. Choose one or more summary statistics to be included in the output table:

- Average
- Sum
- Standard Deviation
- Variance
- AREA_HA
  - Minimum
  - Maximum
  - Average
  - [✓ Sum]
  - Standard Deviation

3. Specify output table: C:\02_ARTIGOS\GODINHO_etal_SMALL-SCALE_FARM

Summarize on the selected records only

About summarizing data

OK Cancel

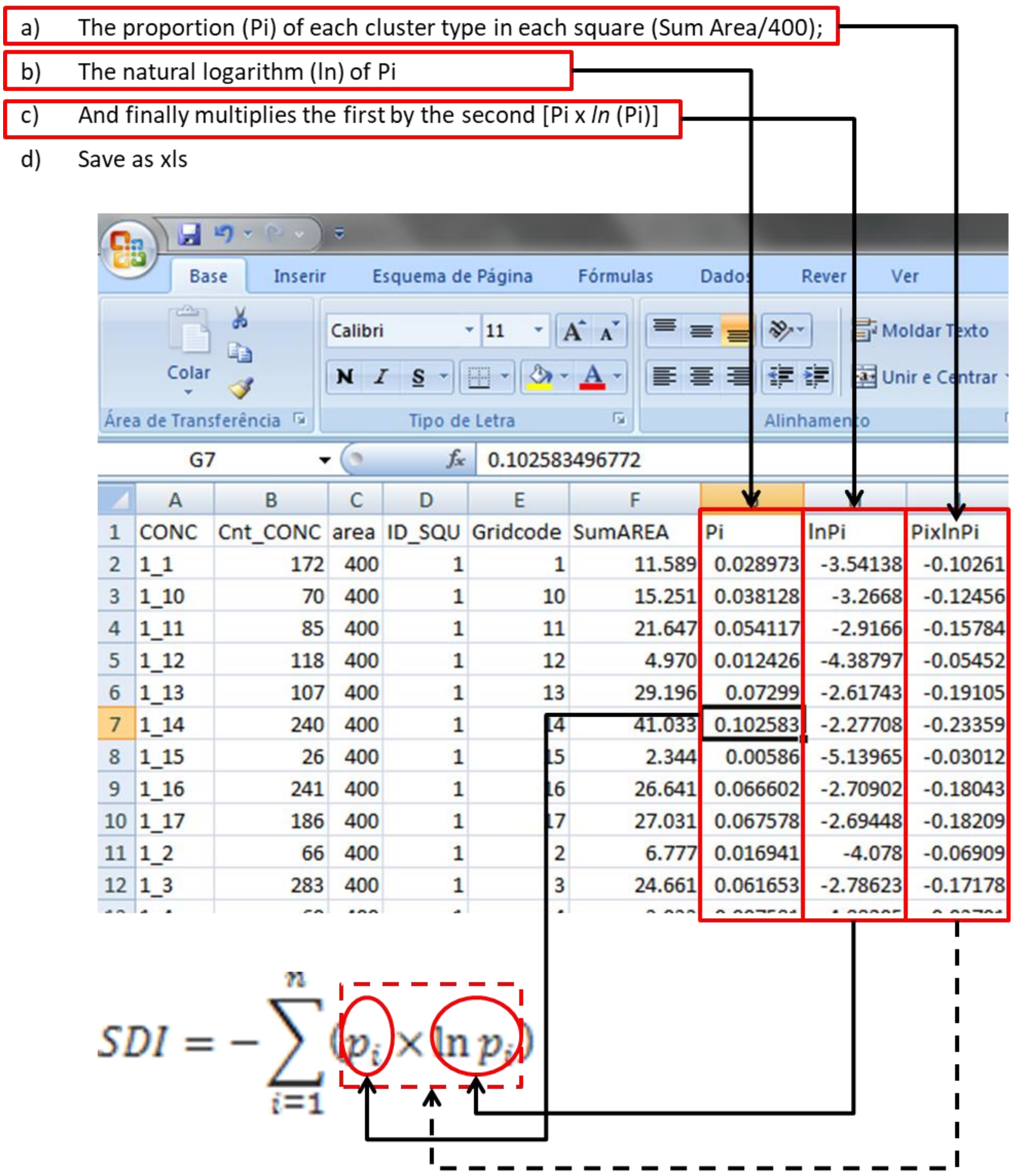
Name: Sum_Output.dbf

Save as type: File and Personal Geodatabase tables

Figure 2 - Build a summary table.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

c) Open the dbf table in Excel and compute p_i an $\ln(p_i)$



a) The proportion (Pi) of each cluster type in each square (Sum Area/400);

b) The natural logarithm (ln) of Pi

c) And finally multiplies the first by the second [Pi x *ln* (Pi)]

d) Save as xls

G7 f_x 0.102583496772

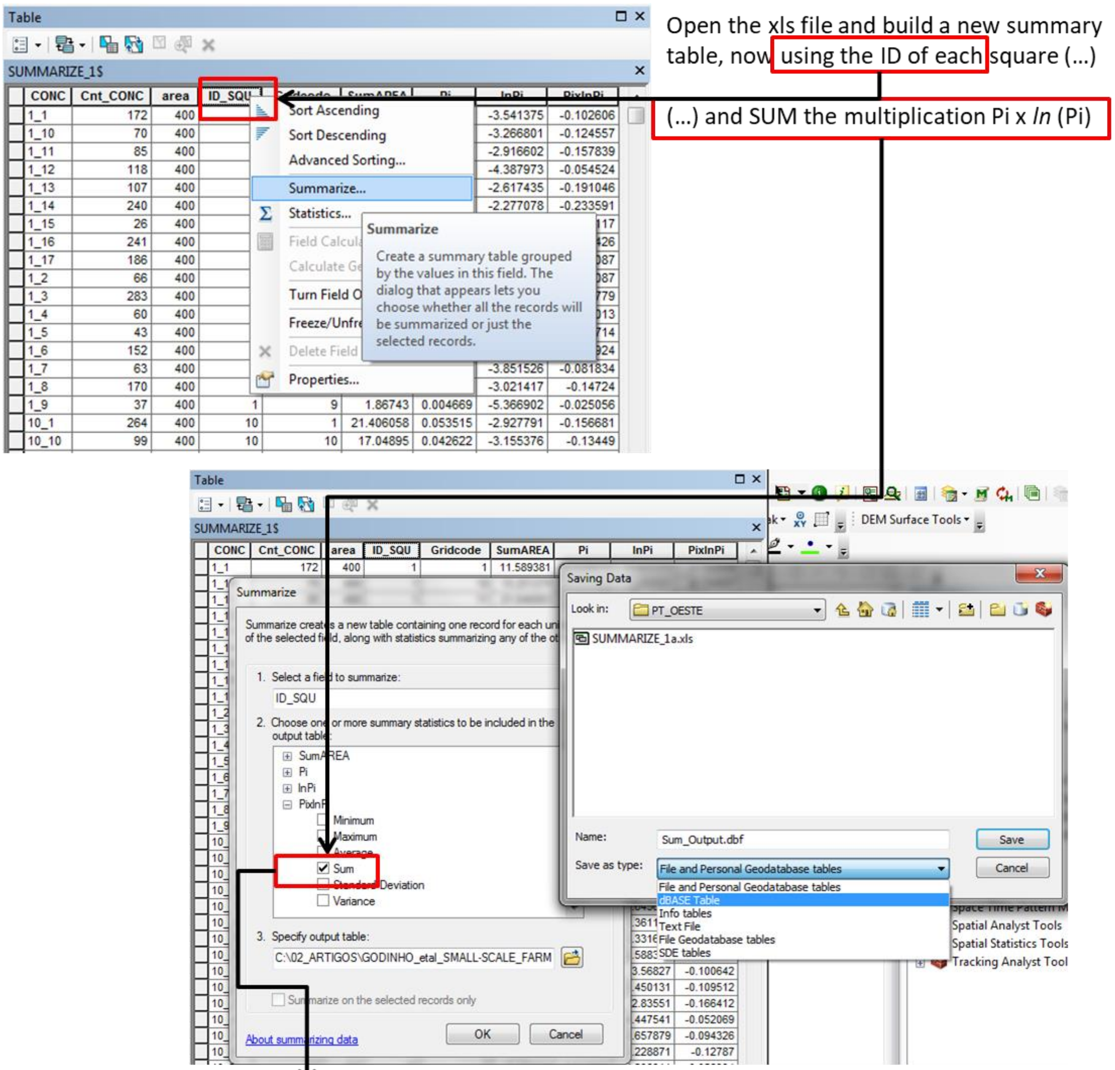
	A	B	C	D	E	F	G	H	I
1	CONC	Cnt_CONC	area	ID_SQU	Gridcode	SumAREA	Pi	lnPi	PixlnPi
2	1_1	172	400	1	1	11.589	0.028973	-3.54138	-0.10261
3	1_10	70	400	1	10	15.251	0.038128	-3.2668	-0.12456
4	1_11	85	400	1	11	21.647	0.054117	-2.9166	-0.15784
5	1_12	118	400	1	12	4.970	0.012426	-4.38797	-0.05452
6	1_13	107	400	1	13	29.196	0.07299	-2.61743	-0.19105
7	1_14	240	400	1	14	41.033	0.102583	-2.27708	-0.23359
8	1_15	26	400	1	15	2.344	0.00586	-5.13965	-0.03012
9	1_16	241	400	1	16	26.641	0.066602	-2.70902	-0.18043
10	1_17	186	400	1	17	27.031	0.067578	-2.69448	-0.18209
11	1_2	66	400	1	2	6.777	0.016941	-4.078	-0.06909
12	1_3	283	400	1	3	24.661	0.061653	-2.78623	-0.17178

$$SDI = -\sum_{i=1}^{n} (p_i \times \ln p_i)$$

Figure 3 - Compute p_i an $\ln(p_i)$ in excel.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

d) Open excel file in ArcGIS and build a new summary table



Open the xls file and build a new summary table, now using the ID of each square (…)

(…) and SUM the multiplication Pi x *ln* (Pi)

SUMMARIZE_1$

CONC	Cnt_CONC	area	ID_SQU	Gridcode	SumAREA	Pi	lnPi	PixlnPi
1_1	172	400					-3.541375	-0.102606
1_10	70	400					-3.266801	-0.124557
1_11	85	400					-2.916602	-0.157839
1_12	118	400					-4.387973	-0.054524
1_13	107	400					-2.617435	-0.191046
1_14	240	400					-2.277078	-0.233591
1_15	26	400						
1_16	241	400						
1_17	186	400						
1_2	66	400						
1_3	283	400						
1_4	60	400						
1_5	43	400						
1_6	152	400						
1_7	63	400					-3.851526	-0.081834
1_8	170	400					-3.021417	-0.14724
1_9	37	400	1	9	1.86743	0.004669	-5.366902	-0.025056
10_1	264	400	10	1	21.406058	0.053515	-2.927791	-0.156681
10_10	99	400	10	10	17.04895	0.042622	-3.155376	-0.13449

Summarize

Create a summary table grouped by the values in this field. The dialog that appears lets you choose whether all the records will be summarized or just the selected records.

CONC	Cnt_CONC	area	ID_SQU	Gridcode	SumAREA	Pi	lnPi	PixlnPi
1_1	172	400	1	1	11.589381			

Summarize

Summarize creates a new table containing one record for each un of the selected field, along with statistics summarizing any of the ot

1. Select a field to summarize:
   ID_SQU
2. Choose one or more summary statistics to be included in the output table:
   - SumAREA
   - Pi
   - lnPi
   - PixlnPi
     - [ ] Minimum
     - [ ] Maximum
     - [ ] Average
     - [x] Sum
     - [ ] Standard Deviation
     - [ ] Variance
3. Specify output table:
   C:\02_ARTIGOS\GODINHO_etal_SMALL-SCALE_FARM

[ ] Summarize on the selected records only

About summarizing data

OK Cancel

Saving Data

Look in: PT_OESTE

SUMMARIZE_1a.xls

Name: Sum_Output.dbf

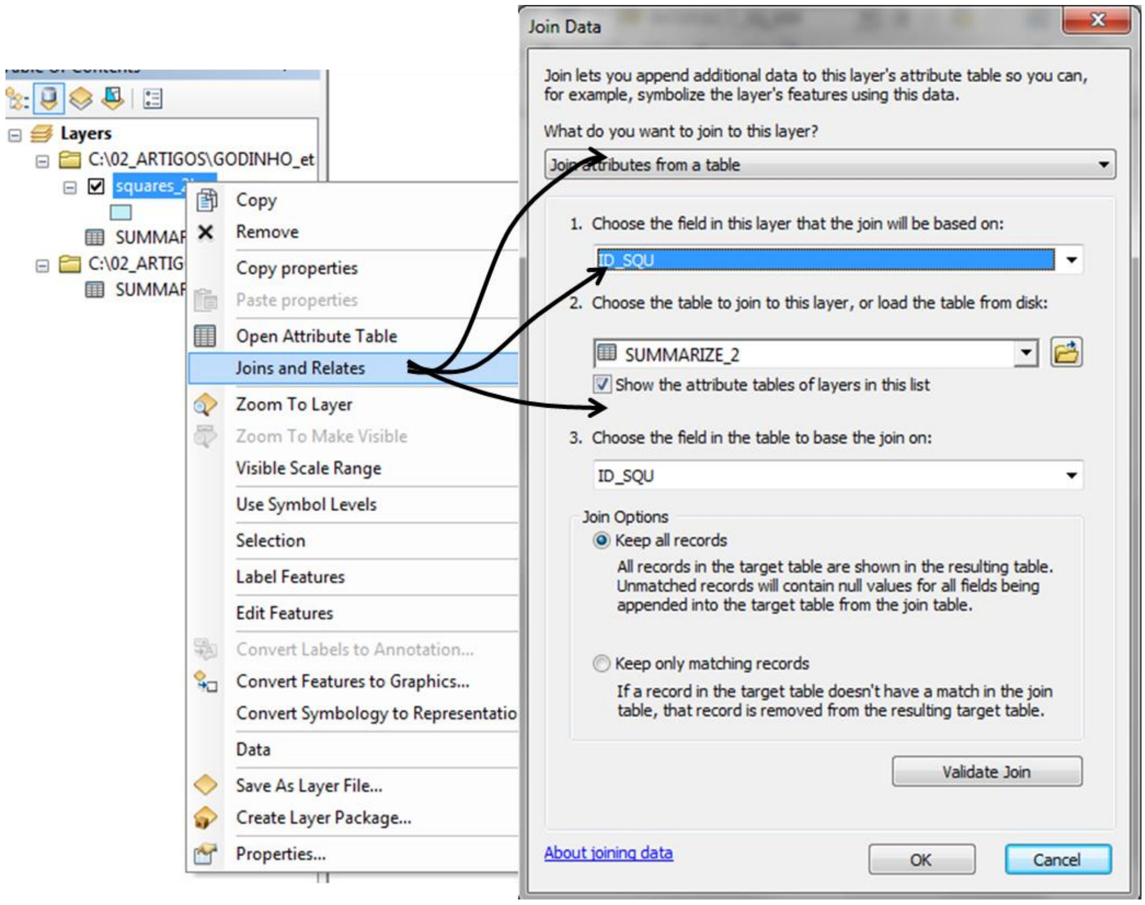
Save as type: File and Personal Geodatabase tables

$$SDI = -\sum_{i=1}^{n} (p_i \times \ln p_i)$$

Figure 4 – Build new summary table.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

e) Join the attributes from the summary table to the 2x2 km squares shapefile using the ID



Join Data

Join lets you append additional data to this layer's attribute table so you can, for example, symbolize the layer's features using this data.

What do you want to join to this layer?

Join attributes from a table

1. Choose the field in this layer that the join will be based on: ID_SQU

2. Choose the table to join to this layer, or load the table from disk: SUMMARIZE_2

[✓] Show the attribute tables of layers in this list

3. Choose the field in the table to base the join on: ID_SQU

Join Options

[●] Keep all records

All records in the target table are shown in the resulting table. Unmatched records will contain null values for all fields being appended into the target table from the join table.

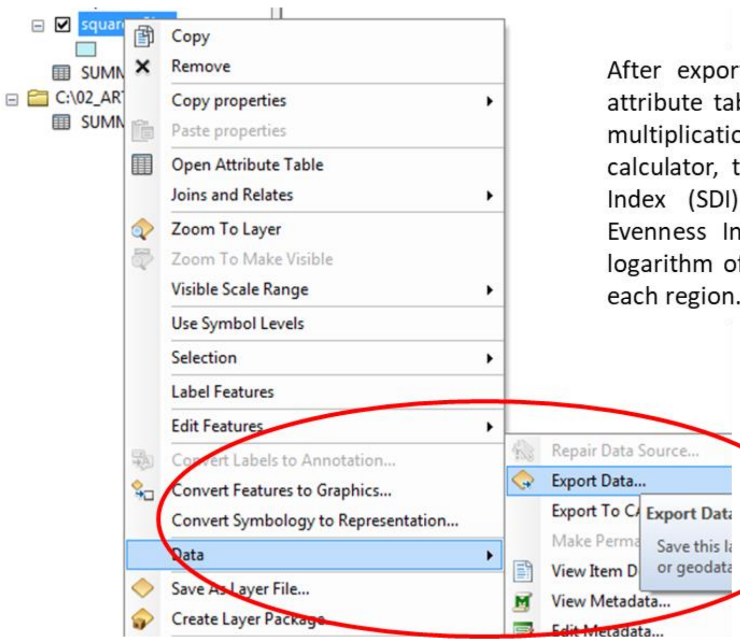
[○] Keep only matching records

If a record in the target table doesn't have a match in the join table, that record is removed from the resulting target table.

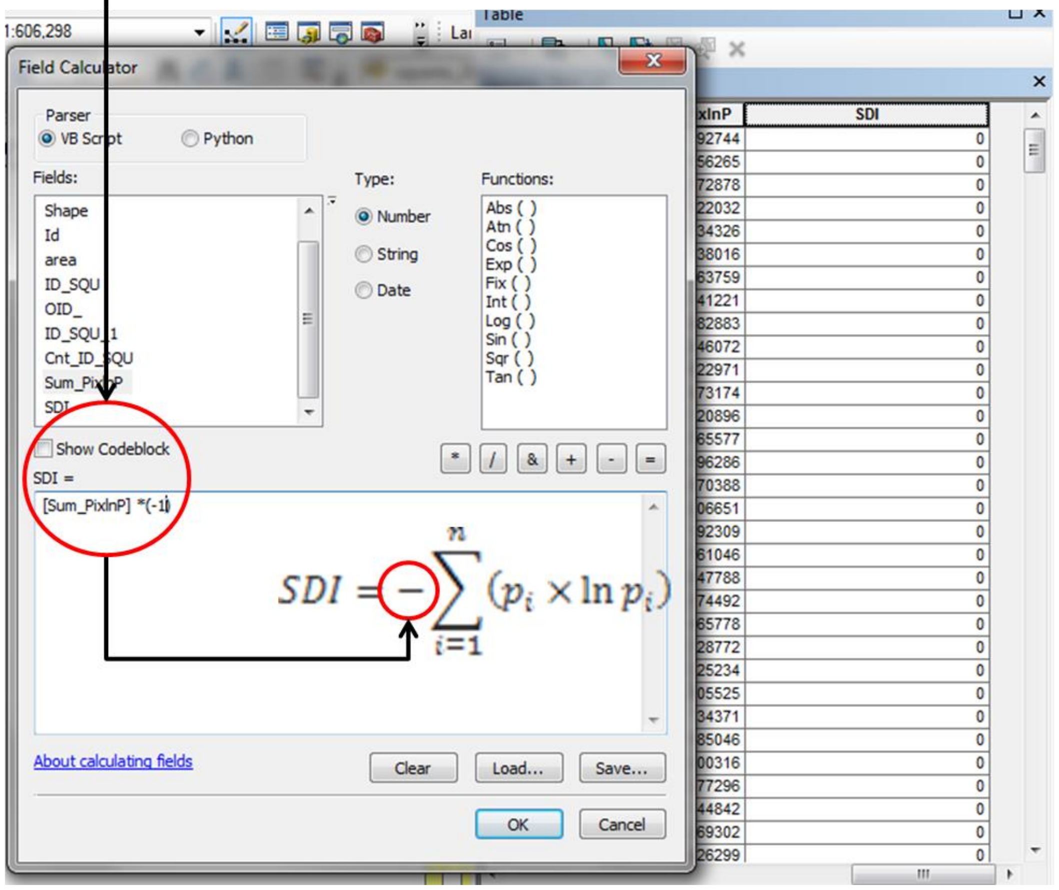
Figure 5 – Join process.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

f) Export to a new shapefile and calculate the Shannon Diversity Index (SDI)



After export data add a new field in the attribute table and multiply the sum of the multiplication by (-1), using the field calculator, to obtain the Shannon Diversity Index (SDI). To determine the Shannon Evenness Index, divide SDI by the natural logarithm of the total number of clusters in each region.



Parser: [✓ VB Script] Python

Type: [✓ Number] String Date

[ ] Show Codeblock

SDI =

[Sum_PixlnP] *(-1)

$$SDI = -\sum_{i=1}^{n} (p_i \times \ln p_i)$$

xlnP	SDI
92744	0
56265	0
72878	0
22032	0
34326	0
38016	0
63759	0
41221	0
82883	0
46072	0
22971	0
73174	0
20896	0
65577	0
96286	0
70388	0
06651	0
92309	0
61046	0
47788	0
74492	0
65778	0
28772	0
25234	0
05525	0
34371	0
85046	0
00316	0
77296	0
44842	0
69302	0
26299	0

Figure 6 - Calculate the SDI and SEI indices.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

4. Compute road distances to assess the degree of accessibility in the squares

Roads from the OpenStreetMaps dataset were used to calculate the Euclidean distance from these linear infrastructures, in order to guarantee a good accessibility to the target squares. Mean distance values within the squares were calculated as follow:

a) Compute distance from roads using Euclidean Distance tool

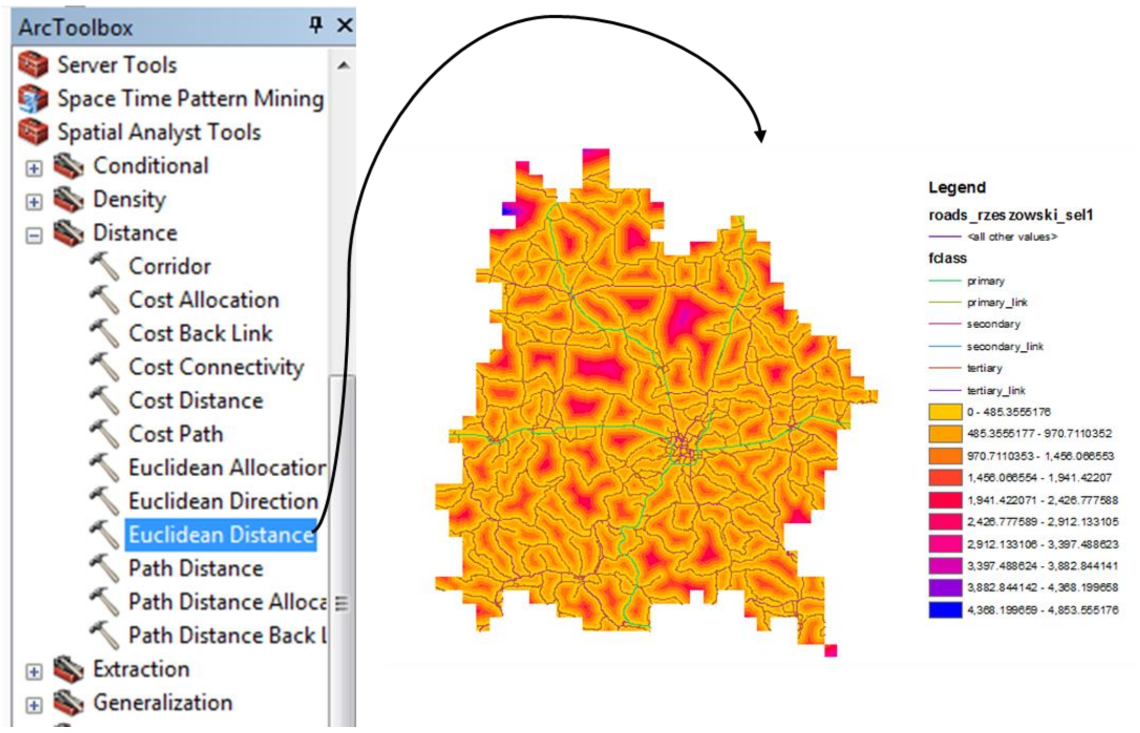
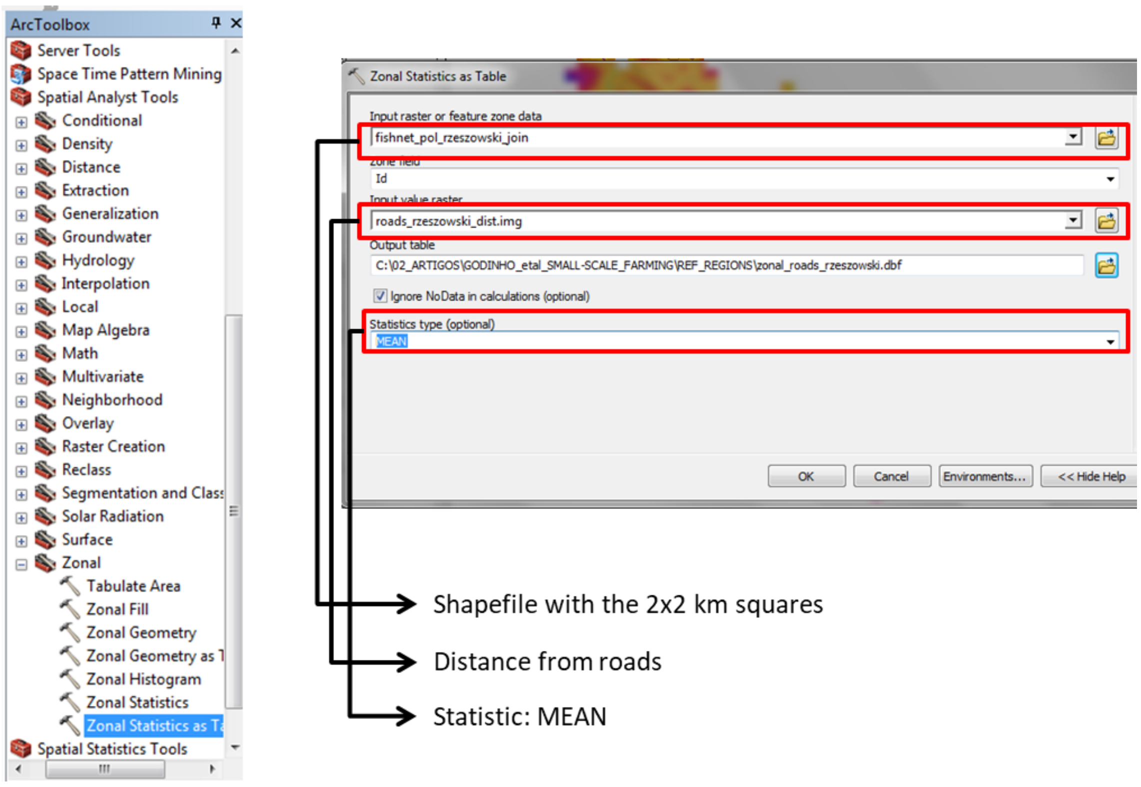


Figure 7 – Compute distance from roads.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

b) Compute the mean distance between roads in each 2x2km square using Zonal Satistics tool.



Input raster or feature zone data: fishnet_pol_rzeszowski_join
Zone field: Id
Input value raster: roads_rzeszowski_dist.img
Output table: C:\02_ARTIGOS\GODINHO_etal_SMALL-SCALE_FARMING\REF_REGIONS\zonal_roads_rzeszowski.dbf
[✓] Ignore NoData in calculations (optional)
Statistics type (optional): MEAN

Figure 8 - Extracting road mean distance per square using zonal statistics tool.

c) Join the resulting table to the shapefile with the 2x2 km squares.

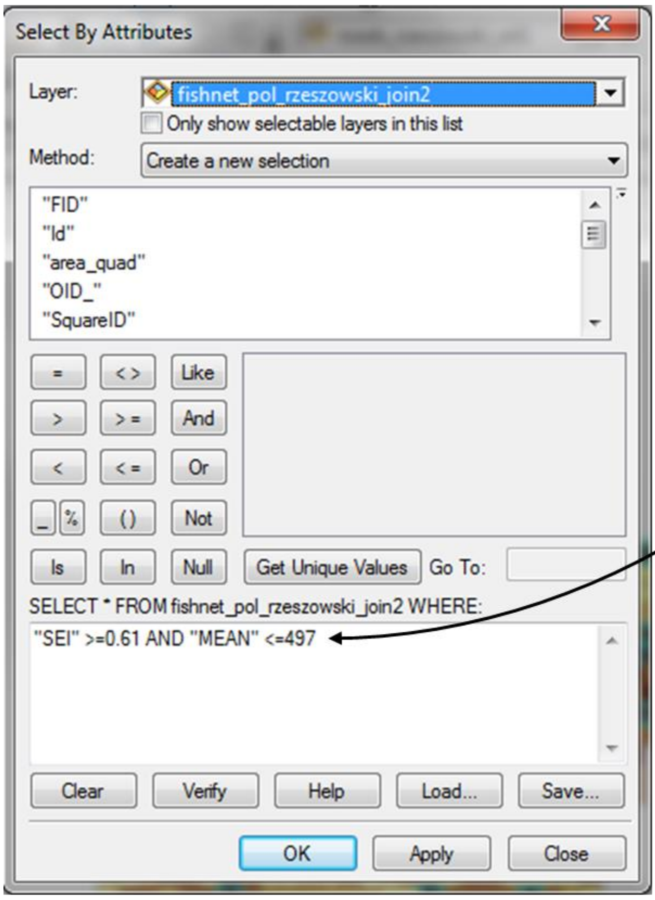
d) Export the shapefile to another one to preserve the join.

5. Squares selection

A previous selection of squares was performed considering the squares with values in the upper quartile of SEI and simultaneously lowers than the mean distance to roads to ensure a selection of squares with high diversity and good accessibility.

To avoid spatial autocorrelation and pseudo-replication problems a minimum distance of 3km between squares was applied, it means that contiguous squares were not selected.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363



Layer: fishnet_pol_rzeszowski_join2

Method: Create a new selection

"FID"
"Id"
"area_quad"
"OID_"
"SquareID"

SELECT * FROM fishnet_pol_rzeszowski_join2 WHERE:

```
"SEI" >=0.61 AND "MEAN" <=497
```

Use selection by attributes to select the squares that have values higher than the upper quartile of SEI and lower than the MEAN (or the MEDIAN) distance to roads: high diversity and good accessibility

Export the selection to a new shapefile. In this exercice (Rzeszowski region) 125 squares were obtained. To avoid spatial auto-correlation a minimum distance of 3 km between squares was applied. Using this criteria the final 25 squares were selected.

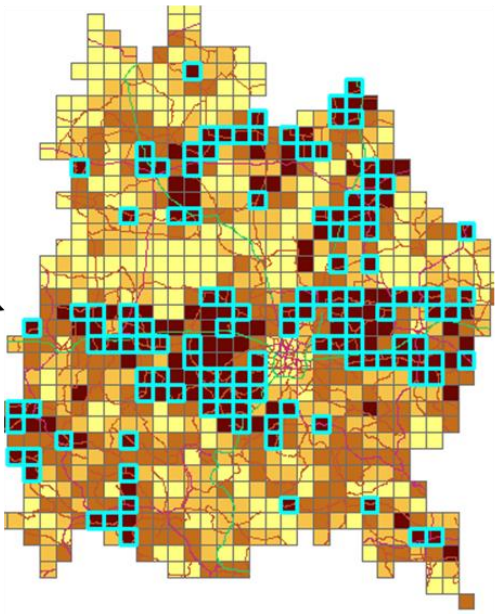


Figure 9 - Final selection.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

6. Points selection

For each one of the 25 squares an average of 20 sample points was distributed along the clusters considering the abundance of each cluster in the square. At a regional level the cluster with the largest area should receive a maximum of 50 points, while the cluster with the lowest area received a minimum of 10 points.

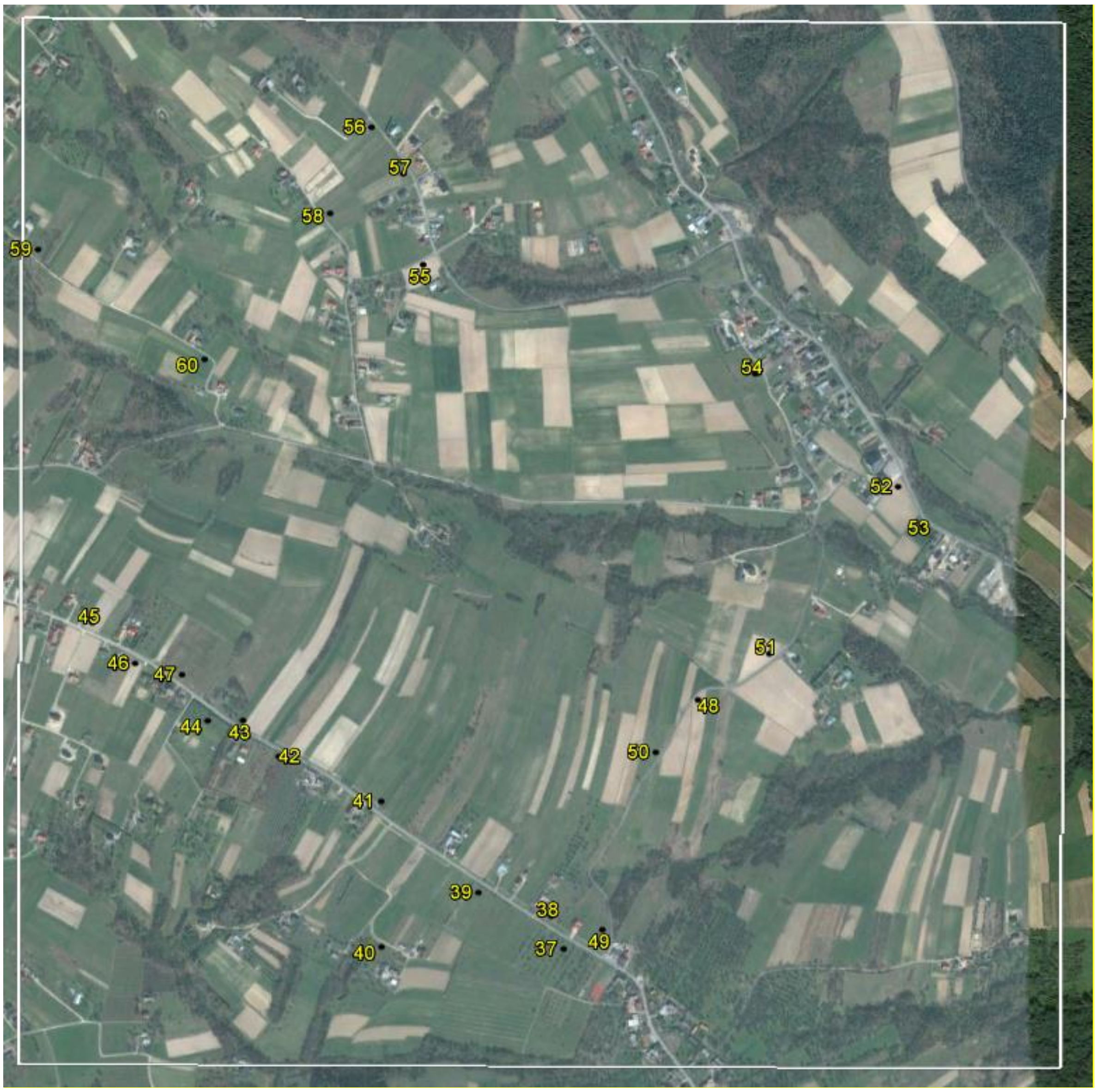


Figure 10 – Example of the spatial distribution of sample points in one 2x2 km square.

ii). quality control of collected reference crop data

With the aim to produce a common output, consistent across reference regions, the assurance of a quality proof product becomes a central issue. The quality control serves as a validation method for all the field information acquired by the corresponding teams in each region. This is crucial to reduce or minimize

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

the error propagation (thematic and geographic corrections) during the entire methodological cycle established to produce the crop type map for each reference region. For the quality control, an average of 8.5% (std = 2.75%) of the sample points (min= 5.1% and max= 15.8%) were checked in 16 out of the 21 references regions. The control points were randomly selected from the previous ~500 points selected for each region. At least one point per square was selected.

iii). Image processing and classification.

This section intends to describe the methodological workflow applied to produce the crop type map. In general this workflow involves 10 main stages to obtain the output 3 (crop type map):

1. Download of multi-temporal Sentinel images and pre-processing,
2. vegetation indices computation to be used as auxiliary information in the classification;
3. create multitemporal NDVI stack for the segmentation stage;
4. image segmentation;
5. prepare the crop dataset to be used in the classification,
6. select the segments intersected by the crop field points;
7. extract all the pixels within the segments to be used in the classification procedure;
8. image classification (pixel based) using Random Forest machine learning algorithm,
9. accuracy assessment of the classification,
10. build the small farms crop type classification map.

Due to the importance and methodological complexity associated with the stages 1, 4-8, and 10 a summary is presented in the following section.

Download of multi-temporal Sentinel images and pre-processing (Stage 1)

For the crop type map classification it was established to use a minimum of two Sentinel-2 images, ideally one from spring and another from summer. However, from the 21 reference regions, and considering the Sentinel-2A and 2B sensors, it was not possible to obtain the minimum cloud-free (<10%) Sentinel-2 images for 6 regions: Montana (Bulgaria); Jihocecky kraj (Czech Republic), Latgale (Latvia), Pieriga (Latvia), Nowosadeki (Poland), and Nowotarski (Poland). Therefore, the solution was to use the Sentinel-1 SAR images for the crop type mapping for those regions. Bellow a brief description about technical characteristics of each satellite, as well as the derived auxiliary variables are presented.

- Sentinel-2

Sentinel-2 is a wide-swath and high-resolution satellite with 13 spectral bands with spatial resolution ranging from 10 m to 60 m. Covering a field of view of 290 km, the 13 spectral bands are collecting information in the visible (VIS), near infrared (NIR) and shortwave infrared (SWIR) wavelengths, with four bands at 10 m, six bands at 20 m and three bands at 60 m spatial resolution (Drusch et al., 2012, Immitzer et al., 2016). For this work only the Sentinel-2 bands at 10 m and 20 m spatial resolution were used, namely the B2 (Blue), B3 (Green), B4 (Red), B5 (Red edge 1), B6 (Red edge 2), B7 (Red edge 3), B8 (NIR1), B8a (NIR2), B11 (SWIR1), and B12 (SWIR2). For the crop type map classification, all the available Sentinel-2 images for each reference region between April and September 2017 were downloaded from the ESA's Sentinel Scientific

This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 677363

Data Hub (https://scihub.copernicus.eu/dhus/). The images correspond to the Level-2A (S2MSI2Ap) processing product which is already atmospherically corrected. Only images with cloud coverage less than 10% were used. After downloading stage the images were processed as follow:

- Clip images using reference region shapefile;
- Band set (to produce a raster file with the 10 S2 clipped bands for each data);
- Images mosaicking;

The integration of multi-seasonal images and vegetation indices into the classification process is a common procedure to increase the spectral separability between different land cover types (e.g., Carrão et al., 2008, Godinho et al., 2016; Rodriguez-Galiano et al., 2012, Senf et al., 2015). This is particularity useful in agricultural landscapes were higher spatial and spectral heterogeneity is a dominant characteristic due to different agricultural plot sizes and high crop diversity. Therefore, in order to increase the inter-class separability four vegetation indices were computed and used as auxiliary variables in the classification, namely, the Normalized Difference Vegetation Index (NDVI), Normalized Difference Water Index (NDWI), Plant Senescent Reflectance Index (PSRI), and shortwave infrared Reflectance 3/2 ratio (SWIR32).

- Sentinel-1

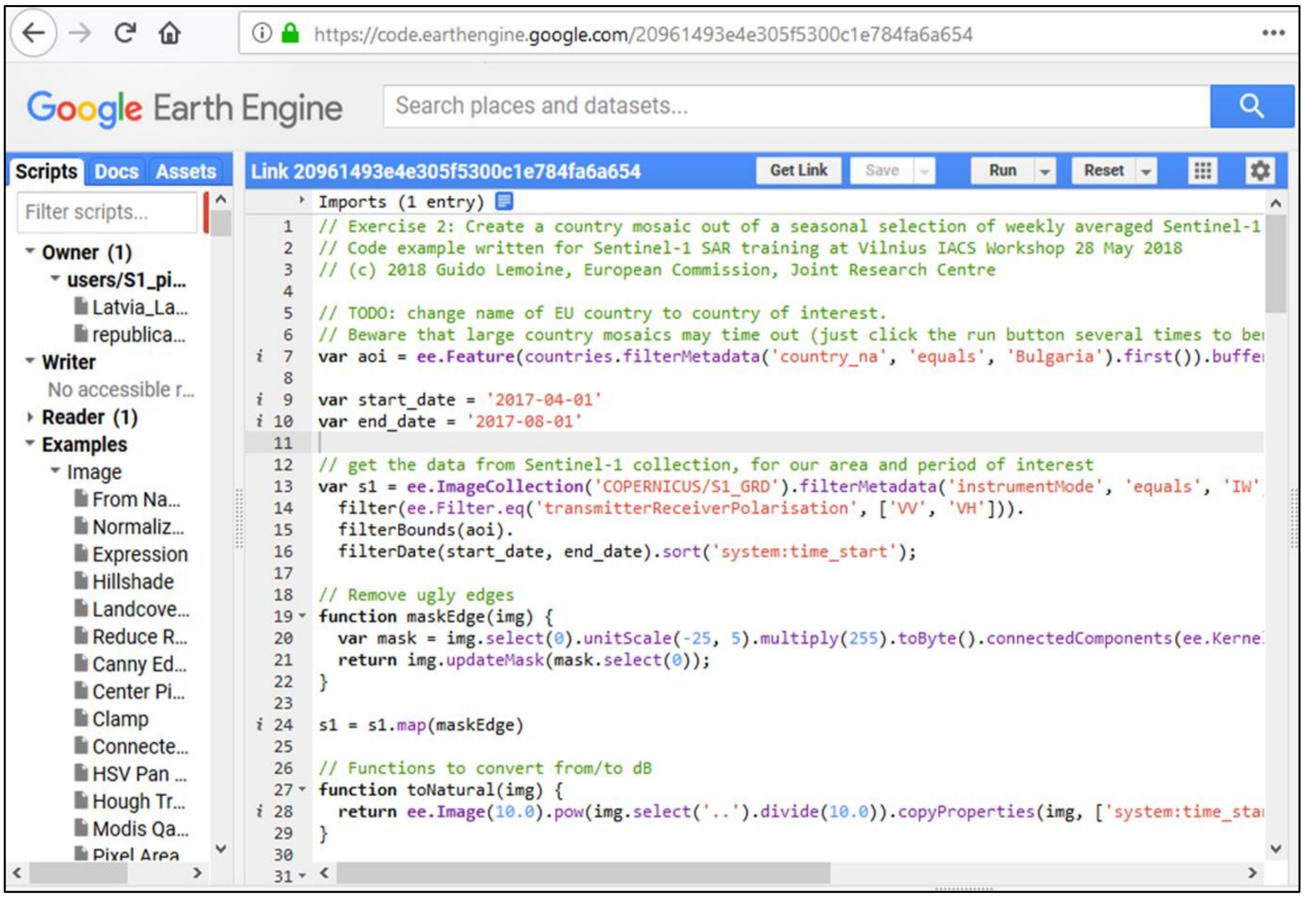
The Sentinel-1 satellite is a radar system with long microwave wavelengths which are able to penetrate clouds. With Sentinel-1A and 1B it is possible to obtain one image for the same place every 6 days, greatly improving the ability to monitor agricultural landscapes throughout an entire growing season. Sentinel-1 is equipped with a C-band imager at 5.405 GHz providing images in dual polarizations (HH+HV, VV+VH) (Torres et al., 2012). Several studies have been showing that VV (transmitter-vertical and receiver-vertical) polarization data is highly influenced by moisture content in soil and vegetation by affecting the absorption and propagation of electromagnetic energy, and thus useful for moisture conditions monitoring (e.g Gao et al., 2018). Regarding VH (transmitter-vertical, receiver-horizontal) polarization data it has been demonstrated that VH is mostly affected by vegetation structure (volume scatter) (e.g. Chaunhan and Srivastava, 2016; Patel et al., 2006). Therefore, in order to increase the accuracy in mapping different crop types, both polarizations were used in this study, as well as different combinations of VV and VH bands ($[VH+VV]^2$, $[VH-VV]^2$, $[VV/VH]$, $[VH/VV]$, $[VV_{min}]$, $[VV_{max}]$, $[VH_{min}]$, $[VH_{max}]$, $[(VV_{july} - VV_{april})/(VV_{july} + VV_{april})]$, $[(VH_{july} - VH_{april})/(VH_{july} + VH_{april})]$). For the period between April and August 2017 the Sentinel-1A and Sentinel-1B backscatter data (descending and ascending passes) in Interferometric Wide Swath mode (IW) at a 10 m spatial resolution were used. The images, Level 1-GRD (Ground Range Detected) product, were downloaded and pre-processed using Google Earth Engine (GEE) cloud computing platform. In GEE Sentinel-1 *ImageCollection* scenes are processed to backscatter coefficient ($\sigma°$) in decibels (dB). S1 Images in GEE are pre-processed following the steps implemented in SNAP Sentinel-1 Toolbox to derive backscatter coefficient for each pixel. These steps are the following:

- Apply orbit file;
- GRD border noise removal;
- Thermal noise removal;
- Radiometric calibration;
- Ortorectification.

Additionally, the Refined Lee speckle filter was applied using 3x3 kernel window. After pre-processing, a weekly mean image was created using all the S1A, S1B images (including ascending and

This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 677363

descending passes) existing within a week. Therefore, from 01/04/2017 to 01/08/2017 17 Sentinel-1 images were produced for each one of the 6 reference regions. The GEE script (Figure 11) used to implement the above mentioned workflow was developed by Dr. Guido Lemoine (European Commission, Joint Research Centre) and can be found here: https://code.earthengine.google.com/20961493e4e305f5300c1e784fa6a654. Note: a GEE account is needed to access the link.



```
// Exercise 2: Create a country mosaic out of a seasonal selection of weekly averaged Sentinel-1
// Code example written for Sentinel-1 SAR training at Vilnius IACS Workshop 28 May 2018
// (c) 2018 Guido Lemoine, European Commission, Joint Research Centre

// TODO: change name of EU country to country of interest.
// Beware that large country mosaics may time out (just click the run button several times to be
var aoi = ee.Feature(countries.filterMetadata('country_na', 'equals', 'Bulgaria').first()).buffe

var start_date = '2017-04-01'
var end_date = '2017-08-01'


// get the data from Sentinel-1 collection, for our area and period of interest
var s1 = ee.ImageCollection('COPERNICUS/S1_GRD').filterMetadata('instrumentMode', 'equals', 'IW'
  filter(ee.Filter.eq('transmitterReceiverPolarisation', ['VV', 'VH'])).
  filterBounds(aoi).
  filterDate(start_date, end_date).sort('system:time_start');

// Remove ugly edges
function maskEdge(img) {
  var mask = img.select(0).unitScale(-25, 5).multiply(255).toByte().connectedComponents(ee.Kerne
  return img.updateMask(mask.select(0));
}

s1 = s1.map(maskEdge)

// Functions to convert from/to dB
function toNatural(img) {
  return ee.Image(10.0).pow(img.select('..').divide(10.0)).copyProperties(img, ['system:time_sta
}

```

Figure 11 – Example of GEE script to produce weekly averaged Sentinel-1 image mosaic.

Overall, for the 21 reference regions, a total of 85 and 61 Sentinel-1 and Sentinel-2 mosaic images were produced, respectively. Table 1 summarizes the number of Sentinel-1 and Sentinel-2 mosaics images generated for each reference region.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

Table 1 – Number of Sentinel-1 and Sentinel-2 mosaic images produced per reference region. Each mosaic represents one different date between April and September 2017.

CODE	COUNTRY	REFERENCE REGION	# of Sentinel-1	# of Sentinel-2	Data amount
R1	Bulgaria	Montana	17	-	13.6 Gb
R4	Czech Rep.	Jihocecký kraj	17	-	38.5 Gb
R6	France	Vaucluse	-	4	10.0 Gb
R8	Greece	Imathia	-	4	12.4 Gb
R9	Greece	Larisa	-	3	29.8 Gb
R10	Greece	Ileia	-	5	20.8 Gb
R11	Italy	Lucca	-	3	8.4 Gb
R12	Italy	Pisa	-	4	18.5 Gb
R14	Latvia	Latgale	17	-	57.2 Gb
R15	Latvia	Pierīga	17	-	28.8 Gb
R16	Lithuania	Vilniaus apskritis	-	4	26.4 Gb
R19	Poland	Rzeszowski	-	2	5.03 Gb
R20	Poland	Nowosadecki	17	-	24.8 Gb
R21	Poland	Nowotarski			
R22	Portugal	Alentejo Central	-	5	25.4 Gb
R23	Portugal	Oeste	-	4	6.36 Gb
R24	Romania	Bistrița-Năsăud	-	3	10.8 Gb
R25	Romania	Giurgiu	-	5	12.9 Gb
R26	Spain	Castellón	-	4	23.0 Gb
R27	Spain	Córdoba	-	5	46.0 Gb
R28	Tunisia	Haouaria	-	6	1.15 Gb
		TOTAL	**85**	**61**	**419.84 Gb**

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

Image segmentation (Stage 4)

Image segmentation was one of the most important stages within the workflow implemented. The goal of this stage was to segment the Sentinel-2 images in polygons that can be used as proxy information about the borders of the agricultural plots. If the images are over segmented, the polygon (segment) size will be very small, and thus the number of false small plots will be extremely higher. On the contrary, if the images are under-segmented the polygons size will be very high, meaning that several small plots were aggregated in one big polygon (Figure 13). These two situations determines significantly the final results because the small plots (< 5ha) will be selected from the segmentation output and used as a proxy of small farms due to the recognized correlation between field crop size and farm size (Fritz et al., 2015; Levin et al., 2006). Image segmentation was performed using Feature Extraction toolbox and the Segment Only Feature Extraction Workflow implemented in ENVI 5.5. For this stage a NDVI layer stack (Figure 12) was created using individual NDVI computed for each S2 date available in each reference region.

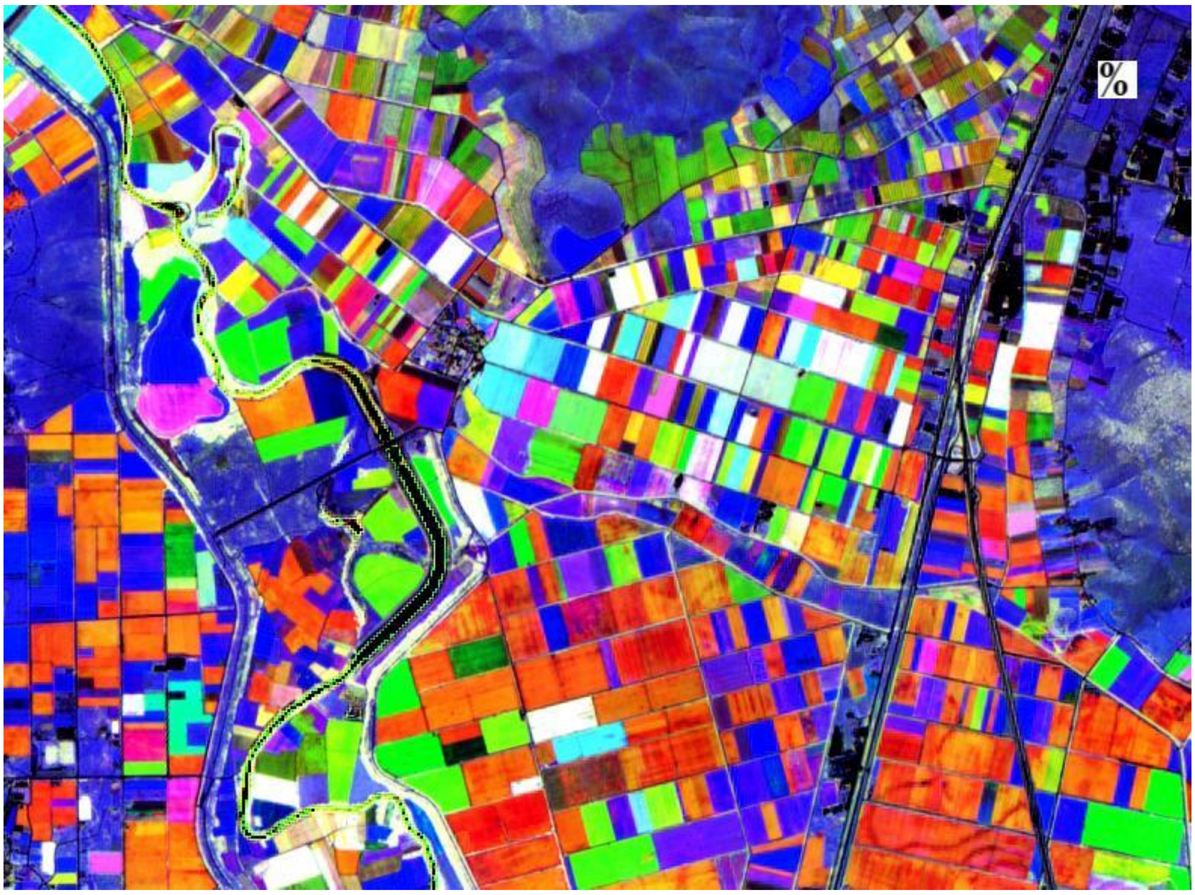


Figure 12 – RGB NDVI composite using different Sentinel-2 images dates. R: $NDVI_{April}$, G: $NDVI_{June}$, B: $NDVI_{August}$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

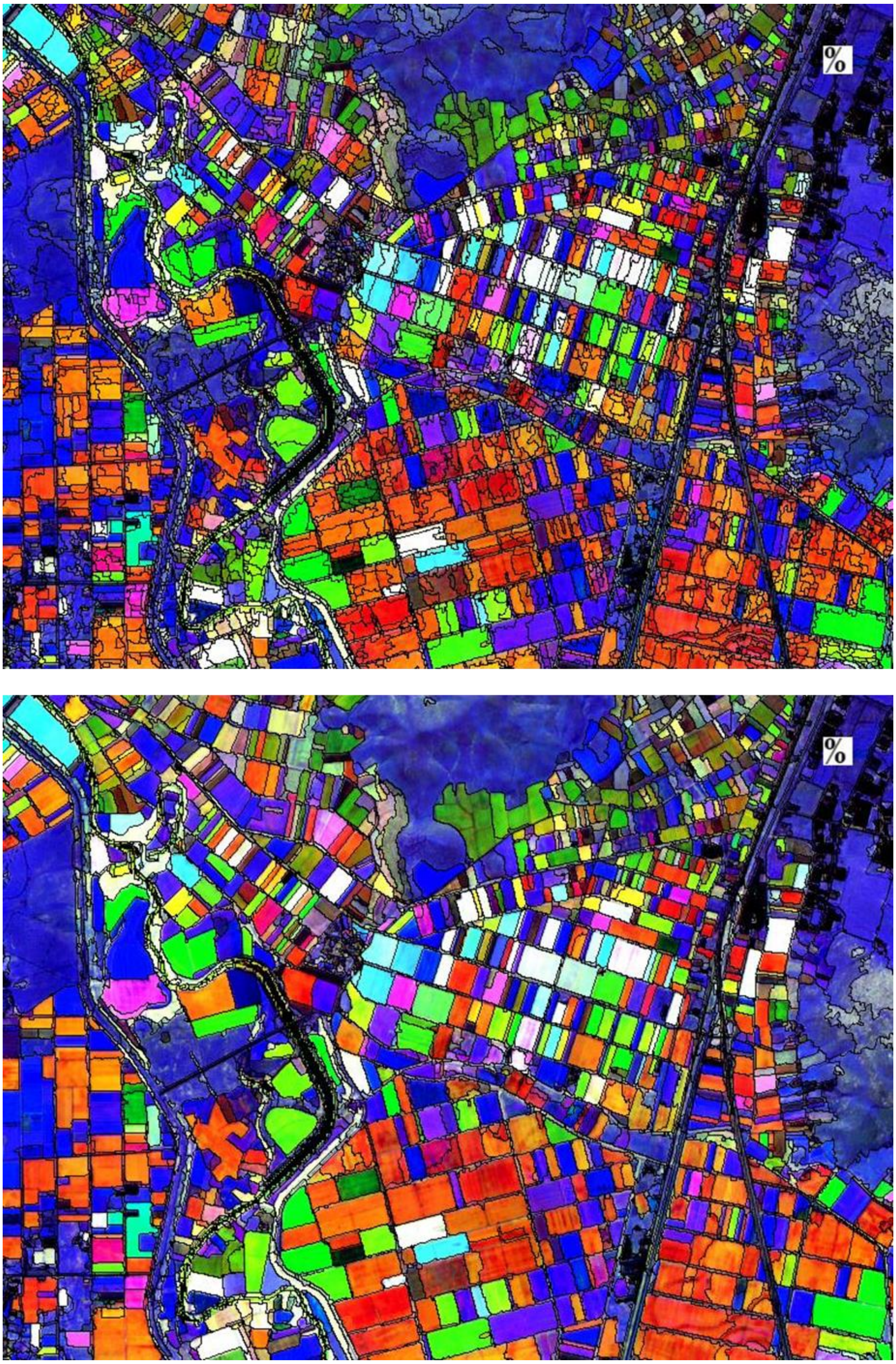


Figure 13 - Upper panel: over segmentation, lower panel: accurate segmentation.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

Crop dataset to be used in the classification (stage 5, 6 and 7)

Based on the crop data collected from both stages of field work (WP3 field points checking and WP2 quality control task), a final crop type dataset for each reference region was built. After the field work all the points were checked one by one through visual inspecting the field digital photos taken by the teams, as well as by superimposing the points over the Sentinel-2 images and very high resolution Google Earth images to check the thematic and geographic accuracy. This procedure was an essential contribution to the high data quality: some points were deleted due to less certainty of the accuracy of the crop identification and exact geographical position (e.g. difficulties in reaching the exact point coordinates). Moreover, all the points coded from field work as “non-identified crop type”, “plowed lands”, and “tillage lands”, were removed from the final dataset.

The flowchart presented in the Figure 14 summarizes the main steps involved in the stage 5, 6 and 7, aiming to generate the final training and testing datasets to be used in image classification and accuracy assessment stages, respectively.

Image classification and accuracy assessment (stage 8 and 9)

A pixel-based supervised Random Forest (RF) machine learning algorithm (Breiman, 2001) was used as a classifier to generate the crop type maps for each reference region. The effectiveness of RF algorithm for remote sensing applications has been demonstrated in several studies (e.g. Freeman et al., 2015; Rodriguez-Galiano et al., 2012; Wang et al., 2016). According to Rodriguez-Galiano et al (2012) this machine-learning algorithm presents many advantages:

- It runs efficiently on large data bases;
- It can handle thousands of input variables without variable deletion;
- It gives estimates of what variables are important in the model;
- It generates an internal unbiased estimate of the generalization error (out-of-bag (oob) error).
- It computes proximities between pairs of cases that can be used in locating outliers.
- It is relatively robust to outliers and noise.
- It is computationally lighter than other tree ensemble methods (e.g. Gradient boosting).

For the RF classification procedure the generated training dataset from stages 5-7 was used. RF classification was implemented using the R package *randomForest*, where only two main tuning parameters need to be parametrized. The first, *mtry*, controls the number of predictor variables randomly sampled to determine each split (Freeman et al., 2015), and $\sqrt{p}$, with *p* the number of predictor variables, was used to determine the value of *mtry*. The second parameter, *ntree*, is the total number of independent trees to grow. The training subset was used to train RF models by using 1000 trees in order to obtain stabilized variable importance estimation (Liaw and Wiemer, 2002).

The test subset was used to evaluate the model performance through the computation of the confusion matrix, which is a cross-tabulation of the crop mapped data against the reference crop data. From the confusion matrix the overall accuracy (OA), the user’s accuracy (UA), and the producer’s accuracy (PA), Omission (OE) and Commission (CE) error were computed. Furthermore, the harmonic mean of the producer and user accuracies, *Fscore*, was computed for each crop using the following equation:

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 677363

$$\mathbf{Fscore} = \frac{\mathbf{2 * PA * UA}}{\mathbf{PA + UA}}$$

Fscore ranges from 0 to one and intends to express the balance between the omission (PA) and commission errors (UA) (Congalton and Green, 2008), the higher the *Fscore* the higher the accuracy classification. An R script was developed to run the random forest classification algorithm, was well as to compute the accuracy metrics (ANNEX II).

Build the small farms crop type classification map (stage 10)

The last stage is dedicated to produce the small plots crop type map. Figure 15 summarizes the main steps implemented to produce this map, which involves the following:

- Intersection between agriculture mask (produced under D.2.2) and segments shapefile to compute the total agricultural area existing in each segment;
- Select segments presenting more than 75% of its area covered by agricultural area and build new shapefile (e.g. agri_segments.shp);
- Intersection between agri_segments.shp with the crop type map generated by random forest (stage 8-9), creating a new shapefile "segment level crop type map.shp";
- Apply a set of generalization processes to determine the dominant crop type existing within each segment;
- Select all the segments with less than 5ha from the above generalized shapefile and build a new shapefile "small plots crop type map.shp".

The generated product from this workflow is the crop type map (output 3) presenting the crop types existing in small plots (5ha) which can be used to compute the crop area as well as the crop production for the selected key crops (according WP3). Crop production is computed by multiplying the total area covered by a specific crop type and the yields of this crop type. Information about the crop yields (ton/ha) was obtained from the small farmer's interviews collected under the scope of WP3.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

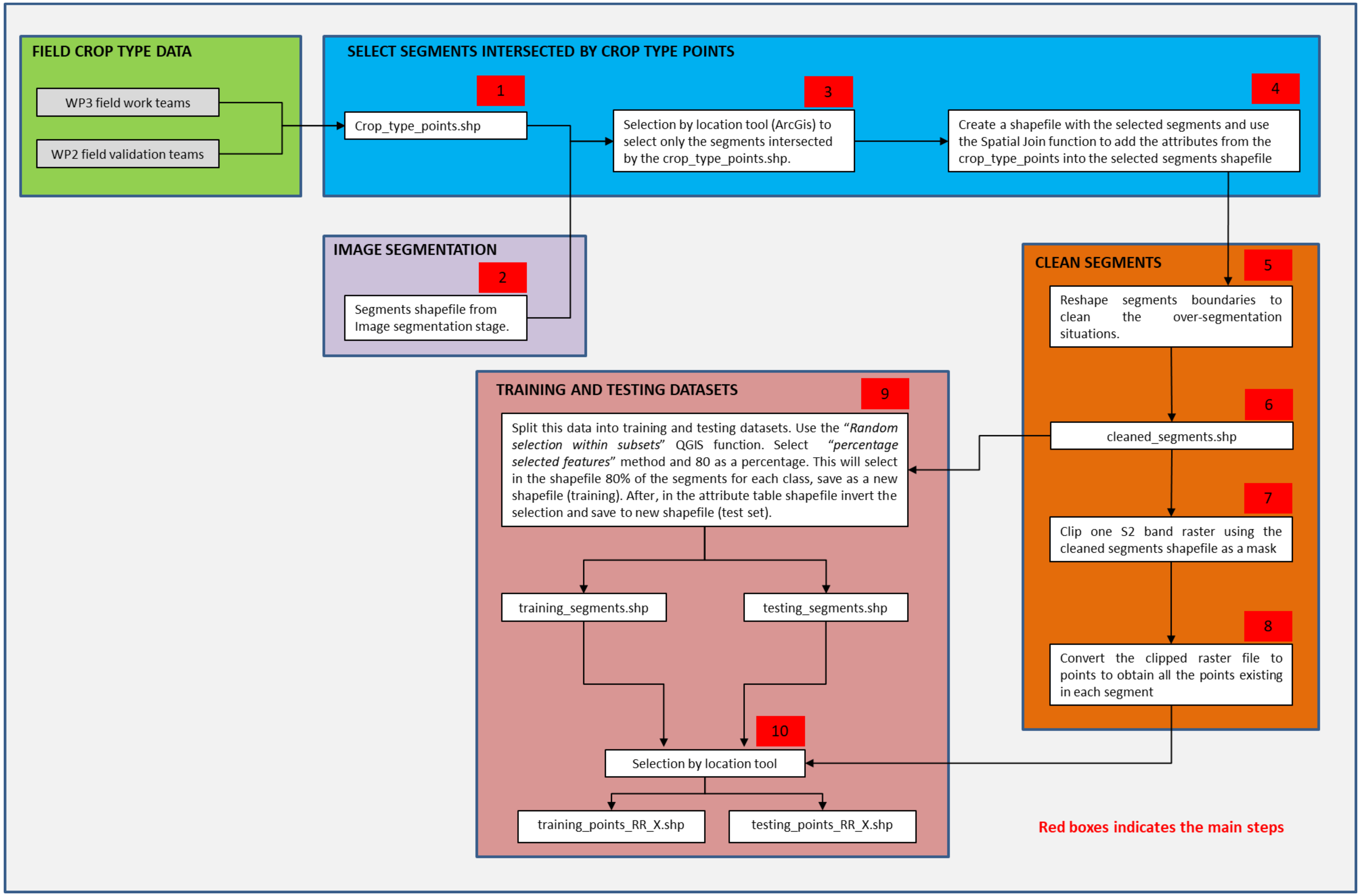

FIELD CROP TYPE DATA
WP3 field work teams
WP2 field validation teams
SELECT SEGMENTS INTERSECTED BY CROP TYPE POINTS
1
Crop_type_points.shp
3
Selection by location tool (ArcGis) to select only the segments intersected by the crop_type_points.shp.
4
Create a shapefile with the selected segments and use the Spatial Join function to add the attributes from the crop_type_points into the selected segments shapefile
IMAGE SEGMENTATION
2
Segments shapefile from Image segmentation stage.
CLEAN SEGMENTS
5
Reshape segments boundaries to clean the over-segmentation situations.
6
cleaned_segments.shp
7
Clip one S2 band raster using the cleaned segments shapefile as a mask
8
Convert the clipped raster file to points to obtain all the points existing in each segment
TRAINING AND TESTING DATASETS
9
Split this data into training and testing datasets. Use the "Random selection within subsets" QGIS function. Select "percentage selected features" method and 80 as a percentage. This will select in the shapefile 80% of the segments for each class, save as a new shapefile (training). After, in the attribute table shapefile invert the selection and save to new shapefile (test set).
training_segments.shp
testing_segments.shp
10
Selection by location tool
training_points_RR_X.shp
testing_points_RR_X.shp
Red boxes indicates the main steps

Figure 14 – Main steps implemented over the stages 5, 6 and 7 to produce the training and testing crop type datasets.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

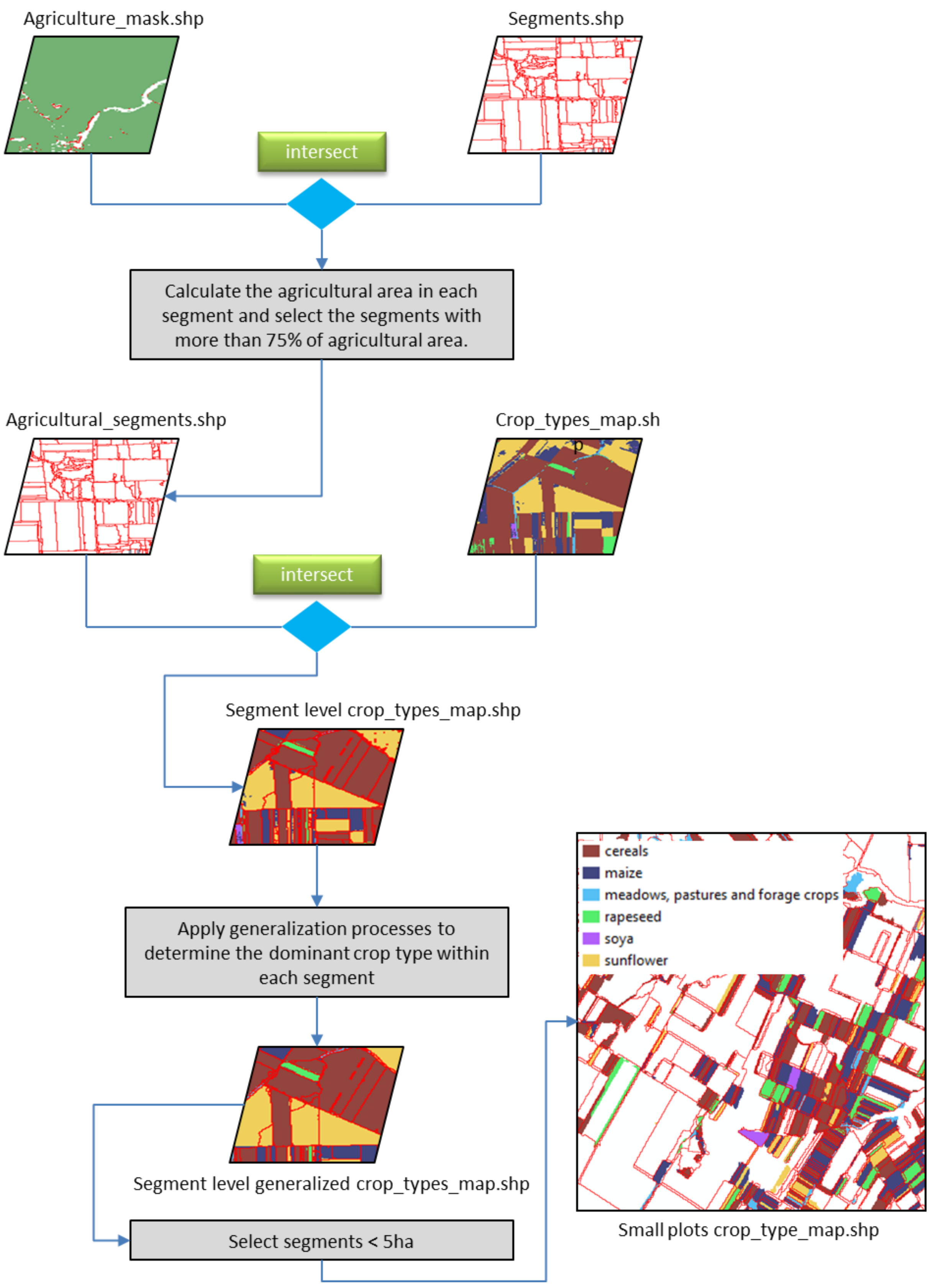

Agriculture_mask.shp
Segments.shp
intersect
Calculate the agricultural area in each segment and select the segments with more than 75% of agricultural area.
Agricultural_segments.shp
Crop_types_map.shp
intersect
Segment level crop_types_map.shp
Apply generalization processes to determine the dominant crop type within each segment
Segment level generalized crop_types_map.shp
Select segments < 5ha
cereals
maize
meadows, pastures and forage crops
rapeseed
soya
sunflower
Small plots crop_type_map.shp

Figure 15 – Main steps to produce the final small plots crop type map.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

3. References

Breiman, L., 2001. Random Forests. Machine Learning, 45:5-32.

Carrão, H., Gonçalves, P., Caetano, M., 2008. Contribution of multispectral and multitemporal information from MODIS images to land cover classification. Remote Sensing of Environment, 112: 986-997.

Chauhan, S., & Srivastava, H.S. 2016. Comparative evaluation of the sensitivity of multi-polarised SAR and optical data for various land cover classes. International Journal of Advancement in Remote Sensing, GIS and Geography, 4: 1 – 14.

Congalton, R.G.; Green, K. Assessing the Accuracy of Remotely Sensed Data: Principles and Practices; CRC Press: Boca Raton, FL, USA, 2008.

Drusch, M., Del Bello, U., Carlier, S., Colin, O., Fernandez, V., Gascon, F., Hoersch, B., Isola, C., Laberinti, P.,Martimort, P., Meygret, A., Spoto, F., Sy, O., F., Marchese, & Bargellini, P., 2012. Sentinel-2: ESA's Optical High-Resolution Mission for GMES Operational Services. Remote Sensing of Environment, 120: 25–36.

Freeman, E. A., Moisen, G. G., Coulston, J. W. and Wilson, B. T. 2015. Random forests and stochastic gradient boosting for predicting tree canopy cover: comparing tuning processes and model performance. Canadian Journal of Forest Research, 45: 1–17.

Fritz, S., See, L., Mccallum, I., et al., 2015. Mapping global cropland and field size. Global Change Biology, 21, 1980–1992.

Gao, Q., Zribi, M., Escorihuela, M.J., Baghdadi, N., Segui, P.Q., 2018. Irrigation Mapping Using Sentinel-1 Time Series at Field Scale. Remote Sensing, 10, 1495.

Godinho, S., Guiomar, N., Gil, A., 2016. Using a stochastic gradient boosting algorithm to analyse the effectiveness of Landsat 8 data for montado land cover mapping: Application in southern Portugal. International Journal of Applied Earth Observation and Geoinformation, 49: 151 – 162.

Immitzer, M., Vuolo, F. and Atzberger, C., 2016. First Experience with Sentinel-2 Data for Crop and Tree Species Classifications in Central Europe. Remote Sensing, 8:166.

Levin, G., 2006. Farm size and landscape composition in relation to landscape changes in Denmark. Danish Journal of Geography, 106, 45–59.

Liaw, A., Wiener, M. , 2002. Classification and regression by randomForest. R News, 2, 18–22.

Patel, P., Srivastava, H.S., Panigrahy, S., Parihar, J.S., 2006. Comparative evaluation of the sensitivity of multi-polarized multi-frequency SAR backscatter to plant density, International Journal of Remote Sensing, 27:2, 293-305.

Rodriguez-Galiano, V.F., Ghimire, B., Rogan, J., Chica-Olmo, M., and Rigol-Sanchez, J.P., 2012. An assessment of the effectiveness of a random forest classifier for land-cover classification. ISPRS Journal of Photogrammetry and Remote Sensing, 67: 93-104.

Senf, C. , Leitão, P.J. , Pflugmacher, D., Linden, S. , Hostert, P., 2015. Mapping land cover in complex Mediterranean landscapes using Landsat: improved classification accuracies from integrating multi-seasonal and synthetic imagery. Remote Sensing of Environment, 156: 527-536.

Torres et al., 2012. GMES Sentinel-1 mission. Remote Sensing of Environment, 120: 9 – 24.

Wang, L., Zhou, X., Zhu, X., Dong, Z. and Guo, W. 2016. Estimation of biomass in wheat using random forest regression algorithm and remote sensing data. The Crop Journal 4: 212–219.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

ANNEX I

R script for Random Forest unsupervised classification

Sérgio Godinho

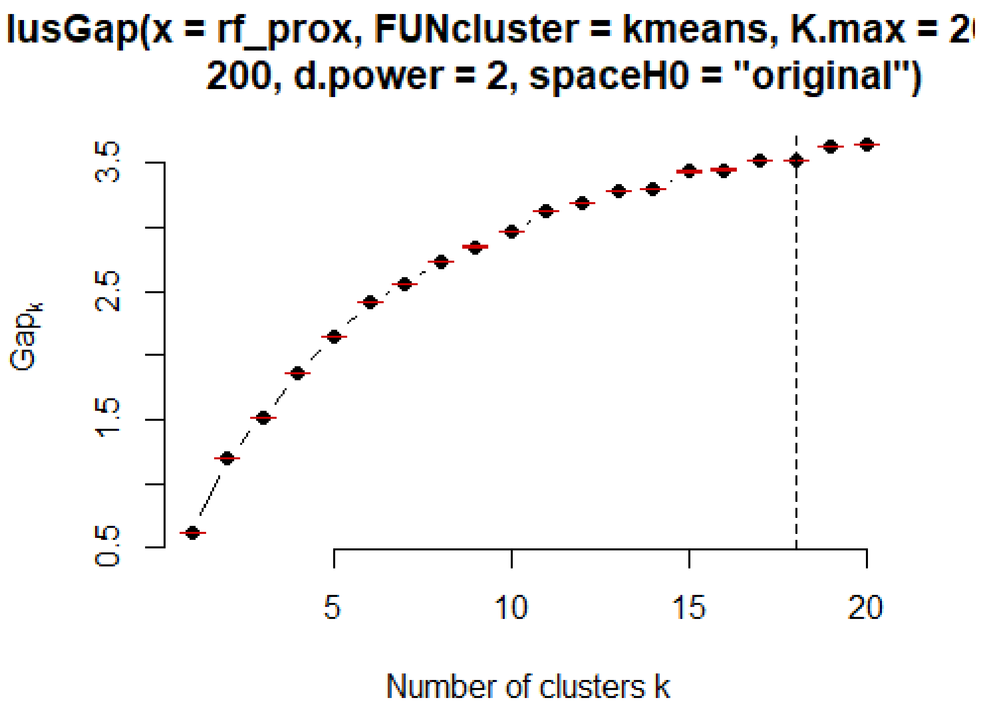
University of Évora

April 2019

```
setwd("D:/R_Vaucluse_RR6")
getwd()

## [1] "D:/R_Vaucluse_RR6"

# UNSUPERVISED RANDOM FOREST CLASSIFICATION ADAPTED FROM “Remote Sensing and GIS for E
cologists” BOOK (edt. by Martin Wegmann, Benjamin Leutner and Stefan Dech). The approa
ch is implemented through three main steps:
# step 1 - use random forest model to calculate proximity values;
# step 2 - these values will be clustered using k-means;
# step 3 - The clusters will be used to train another random Forest model for classifi
cation.

# loading required packages

library(cluster)
library(raster)

library(randomForest)



# IMPORTING RASTER FILES (SENTINEL BANDS) INTO R ENVIRONMENT

# in this case we only used the Sentinel-2 TCI product (true color) to speed up the pr
ocess

rasters_TCI<-stack("TCI_RR6.tif")

# convert the raster data in a matrix and then remove NA-values

v <- getValues(rasters_TCI)
i <- which(!is.na(v))
v <- na.omit(v)

# select a sample of pixels from the matrix "v" created above

vx<-v[sample(nrow(v), 10000),] # in this case we select randomly 10000 from the main m
atrix ()
vx <- na.omit(vx)
vx1<-vx[sample(nrow(vx),1000),] # here we select 1000 cases from the "vx" matrix to us
e in the next step (again to speed up the process)
```

This project has received funding from the European Union’s Horizon 2020
research and innovation programme under grant agreement No 677363

```
# after creating "vx1" object the "v" object should be delected due to its size (could
cause R memory problems), in this example the "v" object presents a 7.8 Gb

remove(v)

# calculating the proximity values

rf_model = randomForest(vx1)
rf_prox <- randomForest(vx1,ntree = 5000, proximity = TRUE)$proximity


# check the "optimal" number of clusters that should be produced with k-means. NOTE: t
his is a very time consuming step.

set.seed(1236)

gap<-clusGap(rf_prox,kmeans,K.max=20,B=500,spaceH0 = "original",d.power = 2)

with(gap, maxSE(Tab[,"gap"], Tab[,"SE.sim"],method="firstSEmax"))

plot(gap,type="b", pch = 19,frame = FALSE, xlab = "Number of clusters k")
abline(v = 18, lty = 2) ## "v" is to indicate the number of clusters obtained
```



```
# calculate the clusters based on the optimal number (in this example the best number
of clusters is 18)

E_rf <- kmeans(rf_prox, 18, iter.max = 500, nstart = 10)
rf_model2 <-randomForest(vx1,as.factor(E_rf$cluster),ntree = 5000,mtry=c(1:3))
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

```
# predict the random forest model (rf_model2) for the raster_TCI object using parallel
processing to produce a map with 18 classes

beginCluster(8)

## Loading required namespace: parallel

preds_rf<-clusterR(rasters_TCI, raster::predict, args = list(model = rf_model2))
endCluster()

# export the map as a tif file

writeRaster(preds_rf,"rf_kmeans_.tif", overwrite= TRUE)
```

This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 677363

ANNEX II

R script for crop type classification using Random Forest and Sentinel data

Sérgio Godinho

University of Évora

April 2019

```
setwd("D:/R_Vaucluse_RR6")
getwd()

## [1] "D:/R_tests_for_reports"

# loading required packages

library(caret)

library(randomForest)

library(raster)

library(sp)
library(rgdal)

library(doParallel)



# IMPORTING RASTER FILES (SENTINEL BANDS) INTO R ENVIRONMENT

list_rasters<-list.files("D:/R_tests_for_reports", pattern=".tif", full.names=TRUE)
rasterslixo5<-stack(list_rasters)

# CALCULATING VEGETATION INDICES FOR EACH SENTINEL-2 IMAGE DATE


# Normalized Difference Vegetation Index (NDVI)
# Normalized Difference Water Index (NDWI)
# Shortwave Infrared Reflectance 3/2 ratio (SWIR32)
# Plant Senescent Reflectance Index (PSRI)

NDWI_apr<-(rasterslixo5$B3_apr-rasterslixo5$B8_apr)/(rasterslixo5$B3_apr+rasterslixo5$
B8_apr)
SWIR32_apr<-(rasterslixo5$B12_apr/rasterslixo5$B11_apr)
PSRI_apr<-(rasterslixo5$B4_apr-rasterslixo5$B3_apr)/rasterslixo5$B6_apr

NDWI_jun<-(rasterslixo5$B3_jun-rasterslixo5$B8_jun)/(rasterslixo5$B3_jun+rasterslixo5$
B8_jun)
SWIR32_jun<-(rasterslixo5$B12_jun/rasterslixo5$B11_jun)
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

```
PSRI_jun<-(rasterslixo5$B4_jun-rasterslixo5$B3_jun)/rasterslixo5$B6_jun

NDWI_ago<-(rasterslixo5$B3_ago-rasterslixo5$B8_ago)/(rasterslixo5$B3_ago+rasterslixo5$
B8_ago)
SWIR32_ago<-(rasterslixo5$B12_ago/rasterslixo5$B11_ago)
PSRI_ago<-(rasterslixo5$B4_ago-rasterslixo5$B3_ago)/rasterslixo5$B6_ago

NDWI_sep<-(rasterslixo5$B3_sep-rasterslixo5$B8_sep)/(rasterslixo5$B3_sep+rasterslixo5$
B8_sep)
SWIR32_sep<-(rasterslixo5$B12_sep/rasterslixo5$B11_sep)
PSRI_sep<-(rasterslixo5$B4_sep-rasterslixo5$B3_sep)/rasterslixo5$B6_sep

# saving as a tif files the above created vegetation indices

writeRaster(NDWI_apr,"NDWI_apr.tif", format="GTiff",overwrite=TRUE)
writeRaster(SWIR32_apr,"SWIR32_apr.tif", format="GTiff",overwrite=TRUE)
writeRaster(PSRI_apr,"PSRI_apr.tif", format="GTiff",overwrite=TRUE)

writeRaster(NDWI_jun,"NDWI_jun.tif", format="GTiff",overwrite=TRUE)
writeRaster(SWIR32_jun,"SWIR32_jun.tif", format="GTiff",overwrite=TRUE)
writeRaster(PSRI_jun,"PSRI_jun.tif", format="GTiff",overwrite=TRUE)

writeRaster(NDWI_ago,"NDWI_ago.tif", format="GTiff",overwrite=TRUE)
writeRaster(SWIR32_ago,"SWIR32_ago.tif", format="GTiff",overwrite=TRUE)
writeRaster(PSRI_ago,"PSRI_ago.tif", format="GTiff",overwrite=TRUE)

writeRaster(NDWI_sep,"NDWI_sep.tif", format="GTiff",overwrite=TRUE)
writeRaster(SWIR32_sep,"SWIR32_sep.tif", format="GTiff",overwrite=TRUE)
writeRaster(PSRI_sep,"PSRI_sep.tif", format="GTiff",overwrite=TRUE)


# CREATE A RASTER LAYER WITH SENTINEL-2 BANDS AND VEGETATION INDICES
# only raster bands and vegetation indices "tif" files should exist in the project dir
ectory, otherwise the layer stack will incorporate all the existing tifs

rasters_list <- list.files(full.names = TRUE, recursive = TRUE, pattern = ".tif")
rasters<- stack(rasters_list)


# CLASSIFICATION MODEL TO CREAT THE CROP TYPE MAP

# importing into R environment the crop type points shapefile ("code_1" is the field c
ontaining the crop type (e.g maize, potatoes, etc.)

points_train<-readOGR(getwd(),"points_train_RR6_finalissimo")

## OGR data source with driver: ESRI Shapefile
## Source: "D:/R_tests_for_reports", layer: "points_train_RR6_finalissimo"
## with 24804 features
## It has 5 fields

points_test<-readOGR(getwd(),"points_test_RR6_finalissimo")
```

This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 677363

```
## OGR data source with driver: ESRI Shapefile
## Source: "D:/R_tests_for_reports", layer: "points_test_RR6_finalissimo"
## with 9137 features
## It has 5 fields

# Extracting raster values to training points shapefile

tab_3<-data.frame(points_train@data,extract(rasters,points_train))

# Excluding from the dataframe the non-useful variables for the classification (e.g. l
ongitude and latitude)

tab_3<-subset(tab_3,select=-c(code_2,code_3,x_coord,y_coord))

# Extracting raster values to testing points shapefile

tab_4<-data.frame(points_test@data,extract(rasters,points_test))

# Excluding from the dataframe non-usefull variables for the classification (e.g. long
itude and latitude)

tab_4<-subset(tab_4,select=-c(code_2,code_3,x_coord,y_coord))

# Creating a formula for Random Forest modelling selecting the desired independent var
iables. In this case independent variables is from column 2 to 56, and dependent is "C
ode_1"

xname<-colnames(tab_3[,2:56])
fmla<-as.formula(paste("code_1~",paste(xname,collapse="+")))

# runing the random forest model using parallel processing mode to speed up the task,h
ere we used ntree= 1000 and mtry= sqroot of the number of independent variables

cluster<-makeCluster(detectCores()-6)
registerDoParallel(cluster)

set.seed(1278)
r_forest<- randomForest(fmla, data=tab_3, ntree = 1000, keep.forest=TRUE,na.action=na.
omit, importance = TRUE, replace=F)

stopCluster(cluster)
registerDoSEQ()

# Checking model´s accuracy by calculating confusion matrix using test samples

classif_crop<-predict(r_forest,newdata=tab_4,type="response")
cm_crop<-confusionMatrix(classif_crop,tab_4$code_1)
cm_crop

## Confusion Matrix and Statistics
##
##                                       Reference
```

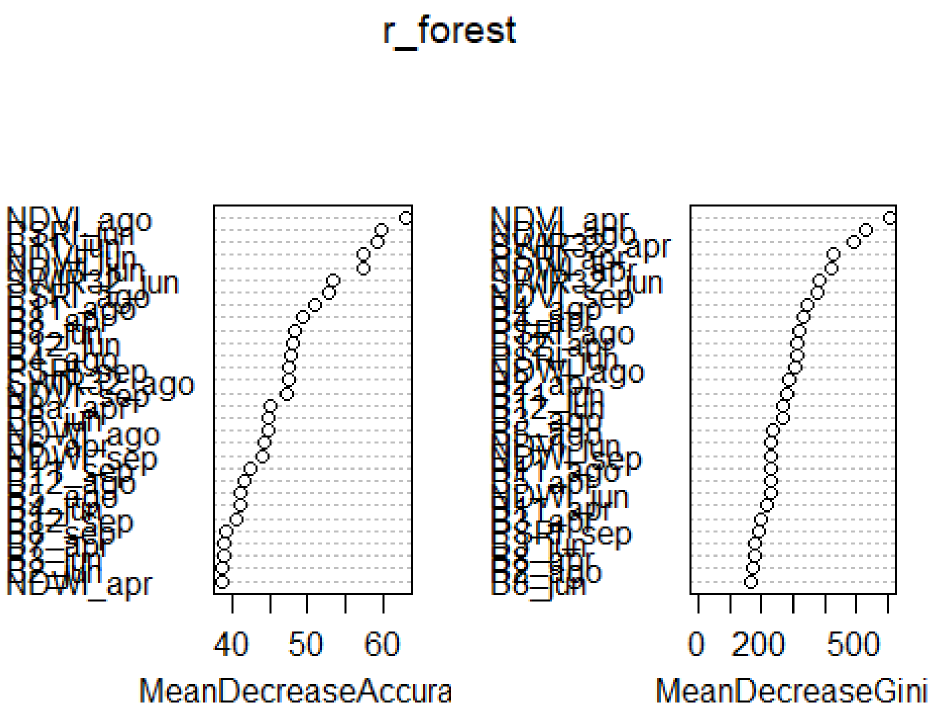
This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 677363

```
## Prediction                            aromatic plants cereal maize
##   aromatic plants                                 930      0     0
##   cereal                                            0   1944     0
##   maize                                             0     88   263
##   meadows, pastures and forage crops                6     31     5
##   olive grove                                       0      0     0
##   orchard                                           0      0     0
##   sunflower                                         2      0     0
##   vegetables                                        0      0    57
##   vineyard                                         76      1     1
##                                        Reference
## Prediction                            meadows, pastures and forage crops
##   aromatic plants                                                      4
##   cereal                                                              81
##   maize                                                                0
##   meadows, pastures and forage crops                                 891
##   olive grove                                                          3
##   orchard                                                              6
##   sunflower                                                            0
##   vegetables                                                           0
##   vineyard                                                            53
##                                        Reference
## Prediction                            olive grove orchard sunflower
##   aromatic plants                               5       0         3
##   cereal                                        6       0         0
##   maize                                         0       0         4
##   meadows, pastures and forage crops           51       9         0
##   olive grove                                   1      38         0
##   orchard                                      10     447         0
##   sunflower                                     0       0       252
##   vegetables                                    0       0         1
##   vineyard                                     80      70        59
##                                        Reference
## Prediction                            vegetables vineyard
##   aromatic plants                              0       22
##   cereal                                       0       95
##   maize                                      103        0
##   meadows, pastures and forage crops           0       16
##   olive grove                                  0       41
##   orchard                                      0        2
##   sunflower                                    4        0
##   vegetables                                 125        0
##   vineyard                                    37     3214
##
## Overall Statistics
##
##                Accuracy : 0.8829
##                  95% CI : (0.8761, 0.8894)
##     No Information Rate : 0.371
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.8481
##  Mcnemar's Test P-Value : NA
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363

```
##
## Statistics by Class:
##
##                      Class: aromatic plants Class: cereal Class: maize
## Sensitivity                          0.9172        0.9419      0.80675
## Specificity                          0.9958        0.9743      0.97787
## Pos Pred Value                       0.9647        0.9144      0.57424
## Neg Pred Value                       0.9897        0.9829      0.99274
## Prevalence                           0.1110        0.2259      0.03568
## Detection Rate                       0.1018        0.2128      0.02878
## Detection Prevalence                 0.1055        0.2327      0.05013
## Balanced Accuracy                    0.9565        0.9581      0.89231
##                      Class: meadows, pastures and forage crops
## Sensitivity                                            0.85838
## Specificity                                            0.98543
## Pos Pred Value                                         0.88305
## Neg Pred Value                                         0.98191
## Prevalence                                             0.11360
## Detection Rate                                         0.09752
## Detection Prevalence                                   0.11043
## Balanced Accuracy                                      0.92191
##                      Class: olive grove Class: orchard Class: sunflower
## Sensitivity                   0.0065359        0.79255          0.78997
## Specificity                   0.9908727        0.99790          0.99932
## Pos Pred Value                0.0120482        0.96129          0.97674
## Neg Pred Value                0.9832118        0.98651          0.99245
## Prevalence                    0.0167451        0.06173          0.03491
## Detection Rate                0.0001094        0.04892          0.02758
## Detection Prevalence          0.0090839        0.05089          0.02824
## Balanced Accuracy             0.4987043        0.89523          0.89464
##                      Class: vegetables Class: vineyard
## Sensitivity                    0.46468          0.9481
## Specificity                    0.99346          0.9344
## Pos Pred Value                 0.68306          0.8950
## Neg Pred Value                 0.98392          0.9683
## Prevalence                     0.02944          0.3710
## Detection Rate                 0.01368          0.3518
## Detection Prevalence           0.02003          0.3930
## Balanced Accuracy              0.72907          0.9412

# Finding the most important varibales

imp<-varImpPlot(r_forest,sort = TRUE)
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 677363



```
# Prediction of r_forest model to obtain the final map

crop_map<-predict(rasters,r_forest,filename="crop_map_RR6.tif",index=1,na.rm=TRUE,type
="class",overwrite=TRUE,progress="window")
```

This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 677363