WATCHITGROW & THE LINK WITH EU EXPLOITATION PLATFORMS

ERWIN GOOR, VITO - 7 SEP 2017 - INSPIRE CONFERENCE STRASBOURG
WatchITgrow®, for the future of the Belgian potato chain
WatchITgrow® for the future of the Belgian potato chain

Monitoring potatoes from space!

» Crop development
» Field heterogeneity
» Risk at production and quality losses
» Yield forecasts

For all actors in the potato chain:
- Get access to satellite images, weather data, yield forecasts
- Store your own field data (e.g. treatments, yield samples,...)
Country wide weather info on a weekly basis:

- Average temperature
- Precipitation sum

“deviation with average”

→ risk at production or quality losses?
Satellite images

» Sentinel-2:
  » 10m pixels
  » Since August 2015
  » Every 10 and soon every 5 days

» DMC/Deimos:
  » 22m pixels
  » since 2009
  » Every 2 days

→ monitor & compare fields
Monitor your fields throughout the season

Emergence →

Senescence →
Check your fields for anomalies

Water logging in June 2016

Sentinel-2 of 20 July 2016 (10m)

UAV image of 18 July 2016 (RGB, 3 cm)

UAV image of 18 July 2016 (NDVI, 8 cm)

Greenness index < 50%: crop lost or severely damaged

www.watchitgrow.be
Check your fields for heterogeneity

- useful for field selection (historical data)
- improved sampling (per zone)
- evolution towards precision agriculture: variable rate application of fertilizers, irrigation, haulm killing,…

**Reference data from soil scans (source: CRA-W)**
Compare your fields

Sentinel-2 of 22 Aug 2016

Senescence started?
Haulm killing applied?

→ optimize field visits
→ input for planning / logistics (harvest)

More advanced senescence
Still green
Yield forecast

Based on combination of yield models
- For 3 varieties: Fontane, Bintje, Nicola
- Per field, municipality, province, region
- From August onwards

Crop growth simulation model

Field data/observations:
- location;
- planting date;
- planting density;
- observations;
- ...

www.watchitgrow.be
Field specific monitoring?

» For farmers
» Data can be shared with other actors in the potato chain

Enter your fields...

- SELECT from existing parcel layer (IACS crop type declarations)
- DRAW your parcel on the map
- IMPORT a shapefile with your parcel boundaries
... and start monitoring your fields!
• View watchITgrow® data and add your own data!

<table>
<thead>
<tr>
<th>Date</th>
<th>No.</th>
<th>Harvest</th>
<th>Plants</th>
<th>Stems per plant</th>
<th>Tubers per plant</th>
<th>Total Fresh Weight</th>
<th>% Dry Matter &gt;35 mm</th>
<th>Under Water Weight &gt;35 mm</th>
<th>% Low Sorting 35-50 mm</th>
<th>% High Sorting &gt;50 mm</th>
</tr>
</thead>
<tbody>
<tr>
<td>13-07-2016</td>
<td>1</td>
<td></td>
<td>7</td>
<td>4.29</td>
<td>16.43</td>
<td>9350 g</td>
<td>18.2 %</td>
<td>329 g</td>
<td>82 %</td>
<td>18 %</td>
</tr>
<tr>
<td>04-08-2016</td>
<td>2</td>
<td></td>
<td>7</td>
<td>4.25</td>
<td>17.71</td>
<td>12350 g</td>
<td>21.2 %</td>
<td>390 g</td>
<td>43 %</td>
<td>57 %</td>
</tr>
<tr>
<td>17-08-2016</td>
<td>3</td>
<td></td>
<td>8</td>
<td>4.88</td>
<td>20.25</td>
<td>15600 g</td>
<td>22.2 %</td>
<td>409 g</td>
<td>42 %</td>
<td>58 %</td>
</tr>
<tr>
<td>21-09-2016</td>
<td>4</td>
<td></td>
<td>7</td>
<td>4.26</td>
<td>15.75</td>
<td>14250 g</td>
<td>22.7 %</td>
<td>420 g</td>
<td>30 %</td>
<td>70 %</td>
</tr>
</tbody>
</table>
Your field from space
PRESENTATION OUTLINE

16 August 2017

proba-v
mep

food security
tep
the Small Satellite for Global Vegetation Monitoring

Copernicus Global Land Service
Global Monitoring of Continental Surfaces
Describe the state, and the dynamics of vegetation, energy budget and water cycle

- User-driven service
- Sustainable system
- Reliable information
- Basic variables
- Validated products
- Near Real Time

Time series since 1999
Global extend

+150 Tbytes/year

Global extend

Proba-VEGETATION

ESA
Belspo

Copernicus space component data access

European Commission

Copernicus
Europe's eyes on Earth
RESTful Web service

https://proba-v-mep.esa.int/api/timeseries/v1.0/ts/PROBAV_L3_S10_TOC_NDVI_333M/?lat=50&lon=4
Welcome to GRASS GIS 7.0.3
GRASS GIS homepage:
This version running through:
Help is available with the command:
see the licence terms with:
If required, restart the GUI with:
When ready to quit enter:
Launching "wxpython" GUI in the back
GRASS 7.0.3 (Global):> }

Space time raster dataset: PROBAV_L3_S10_TOC_NDVI_1Km
Value for 2013-12-01 00:00:00+00:00 is 163
Press Del to dismiss

Temporal resolution: 1 second
Interactive Data Analysis with python & R

Notebooks record & distribute ‘reproducible’ research

```python
# In [1]:
def parameterize(statemap, x, y):
    dataset = gdal.Open(statemap)
    statmap = dataset.GetDefaultBand(1)
    dataset = gdal.Open('example_map(1).tif', 'r')
    return dataset.GetCount(), dataset.GetNbands()

def parameterize2 radiometry, x, y):
    r = gdal.Open(radiometry)
    rastern bands = [r.GetDefaultBand(1).ReadAsArray(x, y) for i in range(r.GetRasterCount())]
    df = pd.DataFrame(rastern bands)
    df = df.reset_index()
    df = df.set_index('x', 'y')
    df = df.T
    return df

def get_dataframe(statemap, x, y):
    dataset = gdal.Open(statemap)
    if dataset.GetCount() == 1:
        array = dataset.GetRasterBand(1).ReadAsArray(x, y)
    else:
        arrays = [dataset.GetRasterBand(i+1).ReadAsArray(x, y) for i in range(dataset.GetRasterCount())]
        array = np.array(arrays)
    return array

# In [2]:
def is_valid(rmap):
    for row in rmap:
        if row[1] == 1:
            return True
    return False

def is_valid(rmap):
    dataset = gdal.Open(rmap)
    if dataset.GetCount() == 1:
        array = dataset.GetRasterBand(1).ReadAsArray()
    else:
        arrays = [dataset.GetRasterBand(i+1).ReadAsArray() for i in range(dataset.GetRasterCount())]
        array = np.array(arrays)
    return array

# In [3]:
def is_valid(rmap):
    dataset = gdal.Open(rmap)
    if dataset.GetCount() == 1:
        array = dataset.GetRasterBand(1).ReadAsArray()
    else:
        arrays = [dataset.GetRasterBand(i+1).ReadAsArray() for i in range(dataset.GetRasterCount())]
        array = np.array(arrays)
    return array

# In [4]:
    def is_valid(rmap):
        dataset = gdal.Open(rmap)
        if dataset.GetCount() == 1:
            array = dataset.GetRasterBand(1).ReadAsArray()
        else:
            arrays = [dataset.GetRasterBand(i+1).ReadAsArray() for i in range(dataset.GetRasterCount())]
            array = np.array(arrays)
        return array

# In [5]:
    def is_valid(rmap):
        dataset = gdal.Open(rmap)
        if dataset.GetCount() == 1:
            array = dataset.GetRasterBand(1).ReadAsArray()
        else:
            arrays = [dataset.GetRasterBand(i+1).ReadAsArray() for i in range(dataset.GetRasterCount())]
            array = np.array(arrays)
        return array

# In [6]:
    def is_valid(rmap):
        dataset = gdal.Open(rmap)
        if dataset.GetCount() == 1:
            array = dataset.GetRasterBand(1).ReadAsArray()
        else:
            arrays = [dataset.GetRasterBand(i+1).ReadAsArray() for i in range(dataset.GetRasterCount())]
            array = np.array(arrays)
        return array
```

Some pixels contain invalid data. In such cases, one of the Reflection value will be equal to -1. Since those pixels contain incomplete data, we might as well filter them out.

As mentioned earlier, the mask for SnowBackground is 100. Since we are only interested in those, we can define a function to convert the complete table into a single bit matrix. If the pixel is SnowBackground and 0 otherwise.

```python
# In [7]:
def is_valid(rmap):
    dataset = gdal.Open(rmap)
    if dataset.GetCount() == 1:
        array = dataset.GetRasterBand(1).ReadAsArray()
    else:
        arrays = [dataset.GetRasterBand(i+1).ReadAsArray() for i in range(dataset.GetRasterCount())]
        array = np.array(arrays)
    return array

# In [8]:
    def is_valid(rmap):
        dataset = gdal.Open(rmap)
        if dataset.GetCount() == 1:
            array = dataset.GetRasterBand(1).ReadAsArray()
        else:
            arrays = [dataset.GetRasterBand(i+1).ReadAsArray() for i in range(dataset.GetRasterCount())]
            array = np.array(arrays)
        return array
```
Register & request a VM / Notebooks access for free!
https://proba-v-mep.esa.int
NextGEOSS
Next generation centralised hub for Earth Observation data and processing

The European contribution to GEO - 10M€, 27 partners, 3.5 years

Next GEOSS Concept

- Provides **access to data** to the European users communities, together with **Cloud resources**, seamlessly connected to provide an integrated ecosystem for supporting applications

- Strong emphasis on **engaging the communities** of European providers and users
The innovative platform aims at simplifying the extraction of information from Earth Observation data for the advancement of data-intensive services in the food security sector mainly in Europe and Africa.

"Supporting Sustainable Food Production from Space"

The innovative platform aims at simplifying the extraction of information from Earth Observation data for the advancement of data-intensive services in the food security sector mainly in Europe and Africa.
Thank you ...

Erwin Goor – VITO - Belgium
erwin.goor@vito.be
+32 493 514274

Presentation ‘Exploitation Platforms in support of Agriculture Monitoring’
Tuesday @ 14h, Room C
Workshop ‘New Ways to Tackle Agriculture Challenges’

Proba-V MEP
Register & request a VM / Notebooks access for free!
https://proba-v-mep.esa.int

WatchItGrow
https://watchitgrow.be