

# Renewable Energy Production Forecast



PROJ0016 - Big Data Project

---

Yann Claes, Gaspard Lambrechts and François Rozet

May 10, 2020

University of Liège

# Photovoltaic production

---



- Maximum power bounding
- Temperature influence
- Sine scaling
- (timezone fixing)



$$P_{out} = \min \left( \frac{\eta I \cos(\theta) A}{\sin(\alpha t)}, \text{MAX\_POW} \right)$$

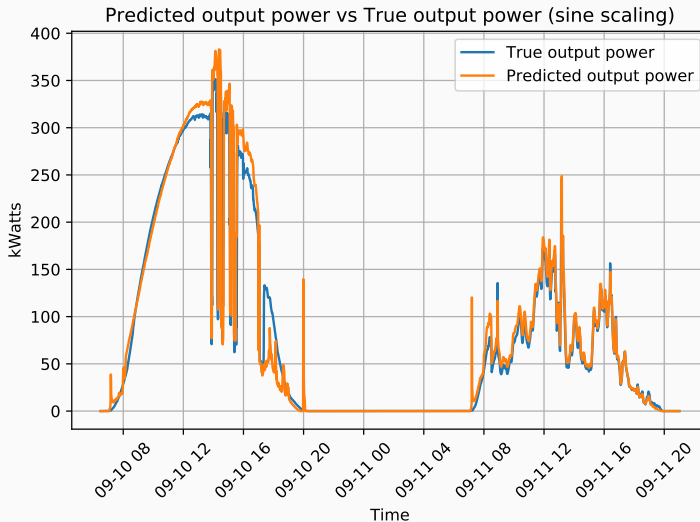
where

$$\text{MAX\_POW} = \min (P_{max}, P_{max} - 0.004(T - 25))$$



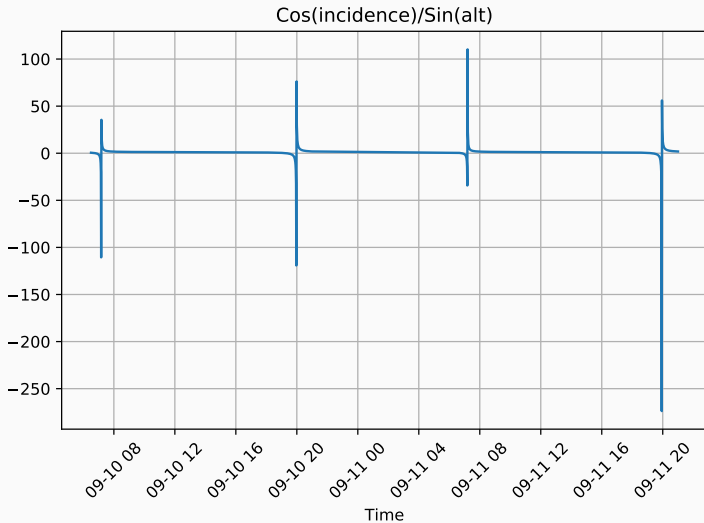
Using the thermodynamics laboratory data, the following model is obtained for a period ranging from September 10<sup>th</sup> to September 12<sup>th</sup>.

# Updated Model Example





There seems to be peaks at regular periods of the day. By looking at the ratio between the sine and cosine, we notice the existence of peaks as well.

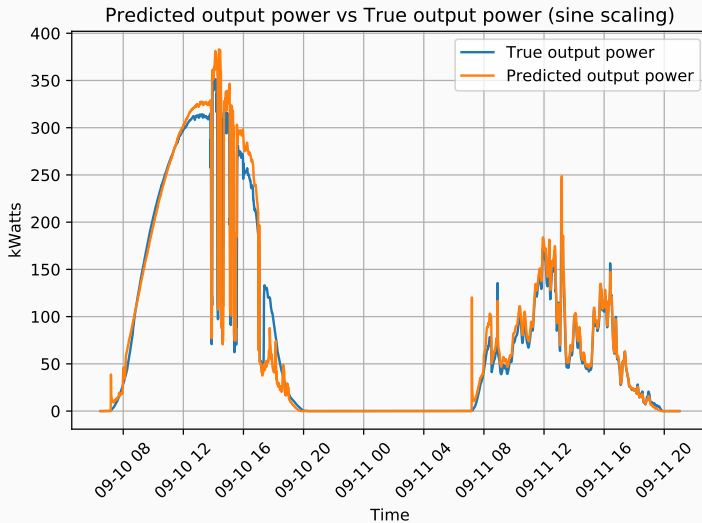






To fix the issue, we decided to force the output power to be equal to zero whenever both quantities are negative: it means the sun is “behind” the panels and has set.

# Final model





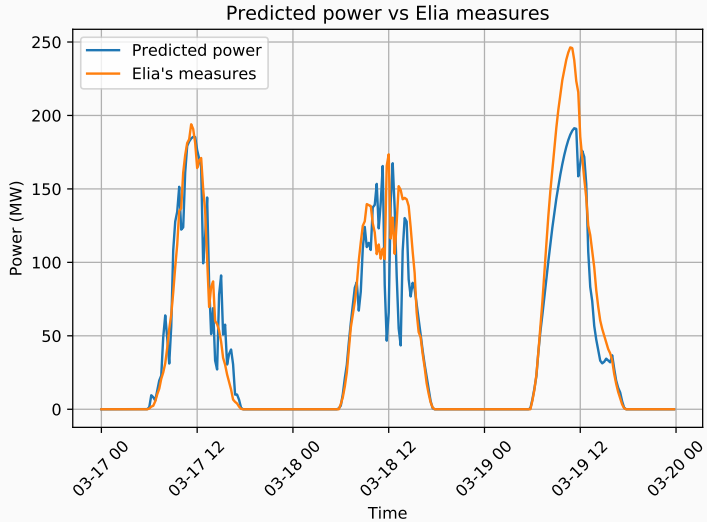
We have access to old (2017-2018) solar statistics data, containing among others the kVA (maximum solar power) of photovoltaic panels installed in the municipalities of Liège.



Using this, as well as some measure of area of photovoltaic panels per kWPeak ( $\sim 7$ ), we obtained a rough measure of the average area of photovoltaic panels installed in the province of Liège.

Setting the efficiency  $\eta$  to some fixed arbitrary value (0.15), we obtain the following model (for a period of 3 days).

# Alternative model





- Use PyStan to fit sensitive parameters to posterior measures made by Elia.
- Potentially use Elia's measure of monitored capacity as starting point



- Deal with uncertainty on parameters ( $\eta$ , peak area, etc.)
- Find relevant irradiance forecast data

# Photovoltaic panels enumeration

---





In the previous review, we talked about using the Google Maps satellite imagery for detection (not training) because of its **high resolution**.

Unfortunately, the Google Maps API **isn't** free. Actually, it is probably reserved to commercial use (cannot remove Google watermark).



Afterwards, we found the **Google Earth Engine** API [1]. It combines a multi-petabyte catalog of satellite imagery and geospatial datasets, like **Landsat** (Nasa) [2] and **Sentinel** (Esa) [3].

The API is primarily available in **JavaScript** and secondarily in **Python**.

The access to this API is restricted, but we obtained the rights.



Unfortunately, the available satellite imagery had an **insufficient resolution** for Belgium.

**WalOnMap** is still our best choice.



Last review, we proposed a few way to investigate in order to find a learning set.

- Asking to a university/repository owner.
- OpenStreetMap solar panel coordinates dataset for England.
- Keep searching for a dataset.



Last review, we proposed a few way to investigate in order to find a learning set.

- Asking to a university/repository owner.
- OpenStreetMap solar panel coordinates dataset for England.
- Keep searching for a dataset.



Last review, we proposed a few way to investigate in order to find a learning set.

- Asking to a university/repository owner. We sent multiple mails, **none were answered**.
- OpenStreetMap solar panel coordinates dataset for England. Gives only (very) approximate location, **not surface nor shape**.
- Keep searching for a dataset. We found one coming from Duke University ! [4]



This dataset contains the geospatial coordinates and border vertices for over 19 000 solar panels across 601 high resolution ( $5000 \times 5000$ ) images from four cities in California (Fresno, Modesto, Oxnard, Stockton).



Figure 1: Modesto city, picture 1, zoomed 10 times





The solar panel  
geospatial coordinates  
are very precise and  
provided as **polygon**  
**vertices** in either a **csv**  
or a **json**.

```
{
  "polygons": [
    {
      "polygon_id": 1,
      "centroid_latitude": 36.926310139710594,
      "centroid_longitude": -119.84055537864529,
      "centroid_latitude_pixels": 107.6184581,
      "centroid_longitude_pixels": 3286.151487,
      "city": "Fresno",
      "area_pixels": 136.192872,
      "area_meters": 1513.254134,
      "image_name": "11ska460890",
      "nw_corner_of_image_latitude": 36.92633611,
      "nw_corner_of_image_longitude": -119.8516222,
      "se_corner_of_image_latitude": 36.91323333,
      "se_corner_of_image_longitude": -119.8343,
      "datum": "NAD83",
      "projection_zone": "11",
      "resolution": 0.3,
      "jaccard_index": 0.914019659,
      "polygon_vertices_lat_lon": [
        [
          -119.84030285285368,
          36.92625126752494
        ],
        [
          -119.84068167074253,
          36.92636019465177
        ]
      ]
    }
  ]
}
```

Figure 2: SolarArrayPolygons.json 20



These coordinates are not useful *as-is* for training.

We have written (using mainly **OpenCV** [5]) a code `learning_set.py` to

1. transform the polygons into black and white **classification** images.
2. slice the images into smaller ( $256 \times 256$ ) images.
3. produce  $(X, Y)$  image pairs.



**Figure 3:** Example of  $(X, Y)$  image pair.



We plan to rotate and translate these  $(X, Y)$  image pairs in order to increase the quantity of available training data.



Now that we have suitable learning and testing sets, we are ready to build our **first detection** model(s).

# Wind production

---



New data has been retrieved:

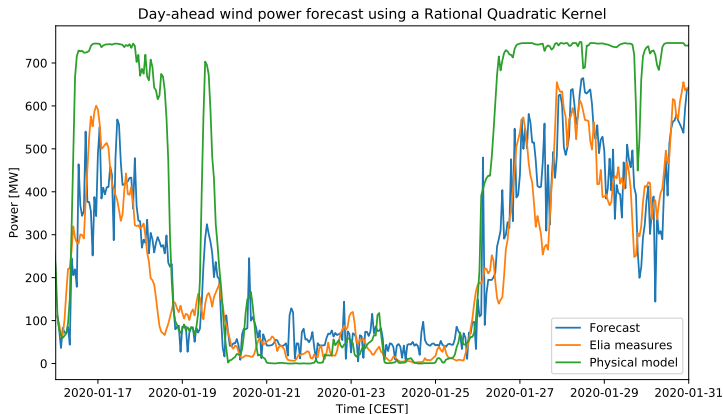
- Wind speed measures from February 2012 to now (one single location in Wallonia for now)
- Physical model output using the wind speed measures and the wind turbines data collected for last review, from 2012 to now
- Elia's wind power measures from February 2012 to now



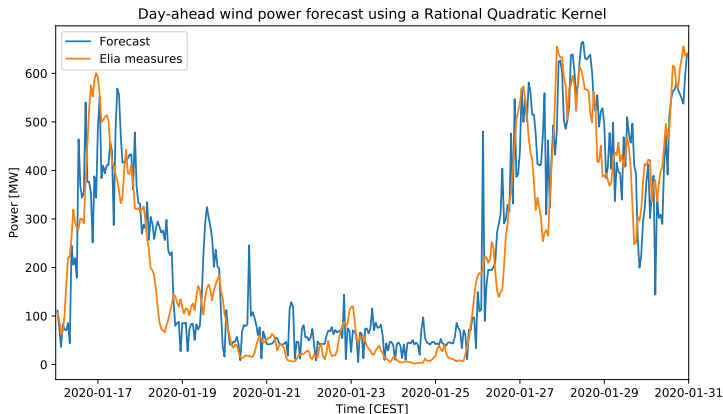
For now, only four types of models have been investigated.

- A **Gaussian Process** based on a Rational Quadratic Kernel predicting the power from the wind speed (at a unique location)
- A **Gaussian Process** based on a Rational Quadratic Kernel predicting **residuals** between the measures of Elia and the physical model
- A **Random Forest** predicting the power from the wind speed, physical model, and timestamp.
- **Extra-Trees** predicting the power from the wind speed, physical model and timestamp.

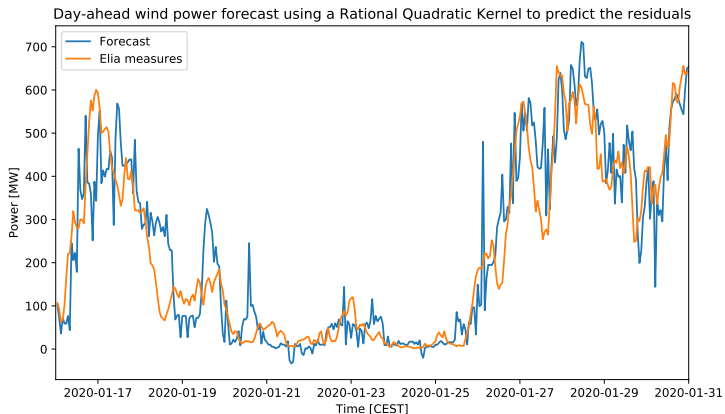




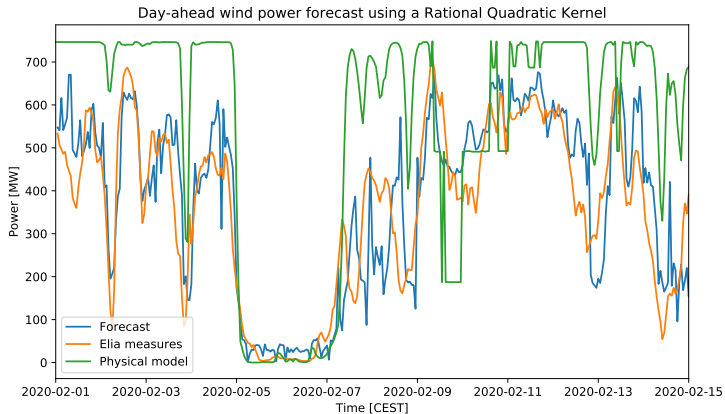
**Figure 4:** Gaussian Process with Rational Quadratic Kernel - 1 month of training data



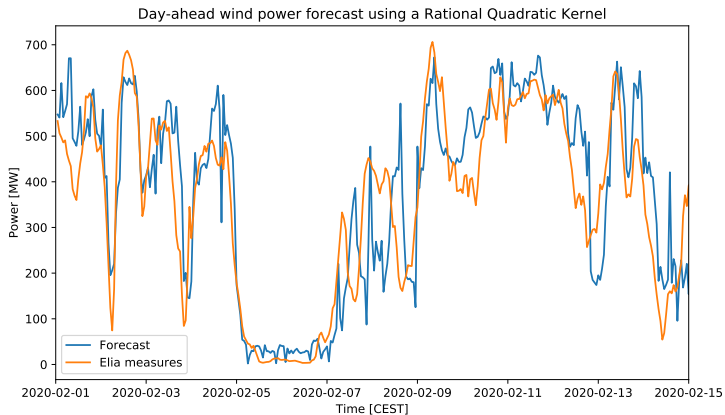
**Figure 5:** Gaussian Process with Rational Quadratic Kernel - 1 month of training data



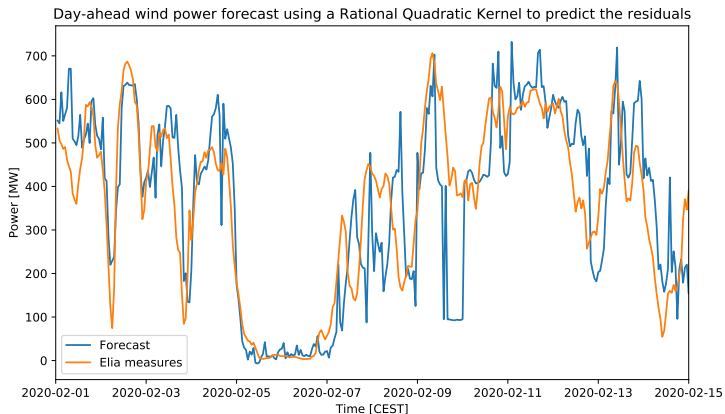
**Figure 6:** Gaussian Process with Rational Quadratic Kernel - 1 month of training data



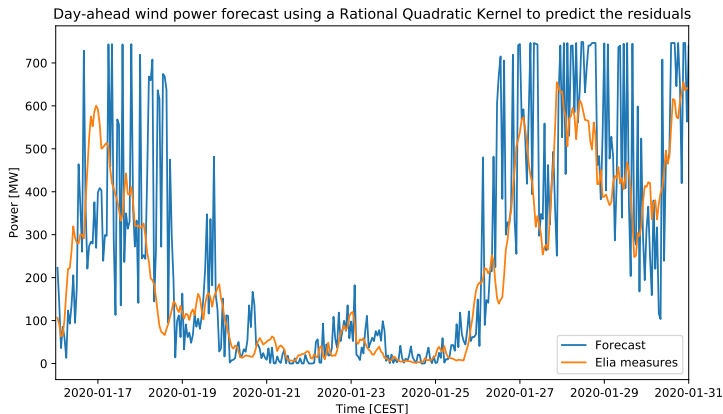
**Figure 7:** Gaussian Process with Rational Quadratic Kernel - 1 month of training data



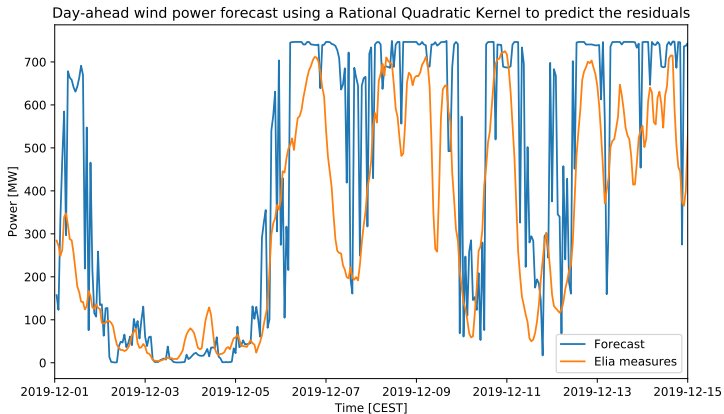
**Figure 8:** Gaussian Process with Rational Quadratic Kernel - 1 month of training data



**Figure 9:** Gaussian Process with Rational Quadratic Kernel - 1 month of training data



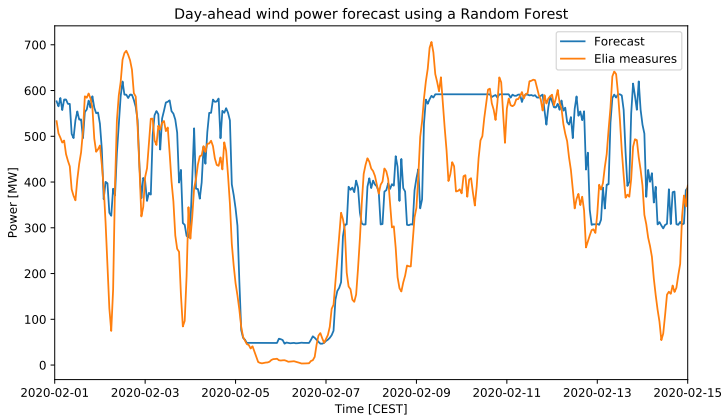
**Figure 10:** Gaussian Process with Rational Quadratic Kernel - 3 months of training data



**Figure 11:** Gaussian Process with Rational Quadratic Kernel - 1 month of training data

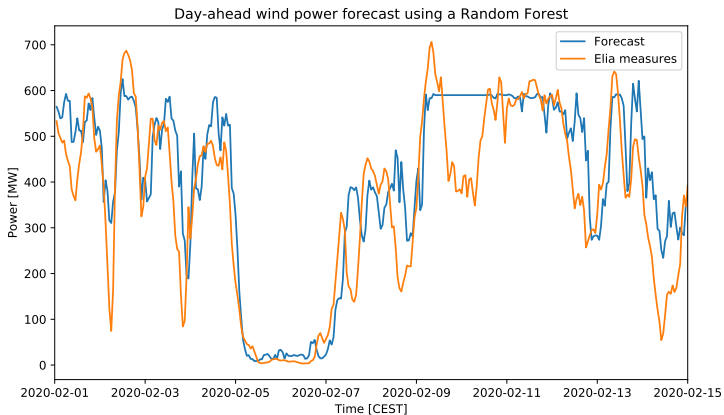


# Result - Random Forest



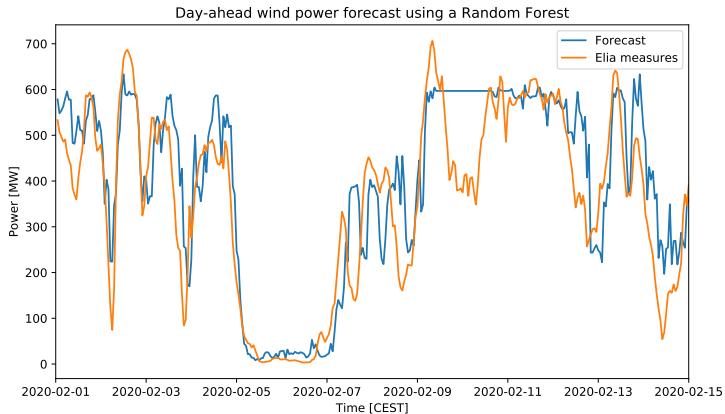
**Figure 12:** Random Forest - 1 month of training data

# Result - Random Forest

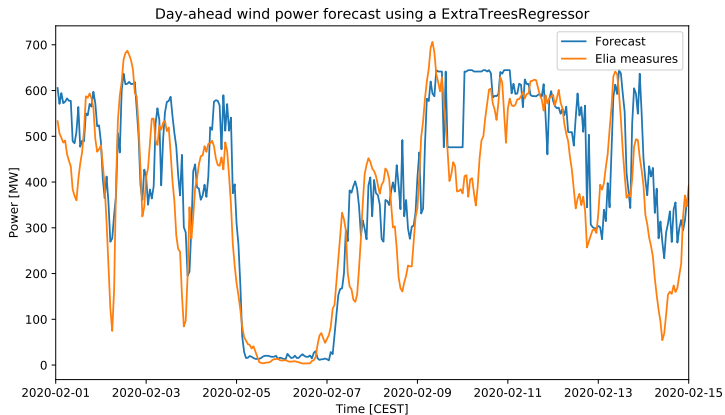


**Figure 13:** Random Forest - 1 year of training data

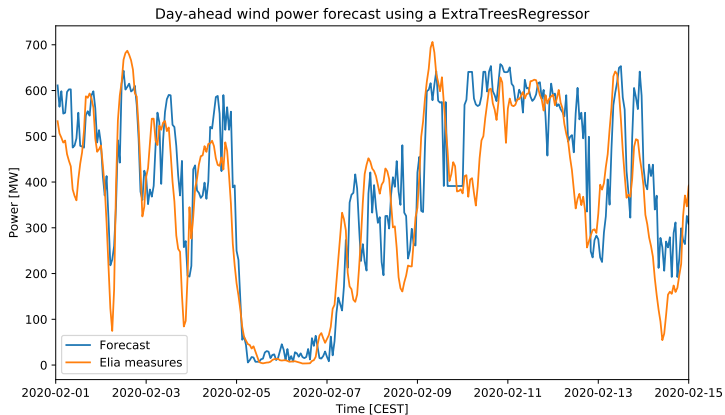
# Result - Random Forest



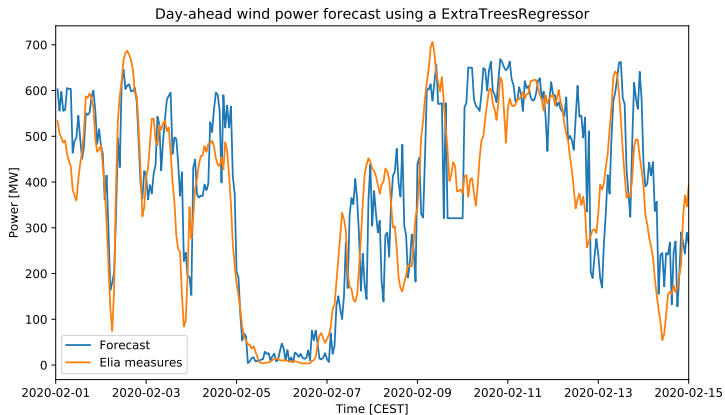
**Figure 14:** Random Forest - 8 years of training data



**Figure 15:** Extra Tree - 1 month of training data



**Figure 16:** Extra Tree - 1 year of training data



**Figure 17:** Extra Tree - 8 years of training data

Questions ?

# References

---



Google. *Google Earth Engine*. 2020. URL:  
<https://earthengine.google.com/>.



Nasa. *Landsat*. URL:  
<https://landsat.gsfc.nasa.gov/>.



Esa. *Sentinel*. URL:  
<https://sentinel.esa.int/web/sentinel/home>.





Kyle Bradbury. *Distributed Solar Photovoltaic Array Location and Extent Data Set for Remote Sensing Object Identification*. 2016. URL:

[https://figshare.com/collections/Full\\_Collection\\_Distributed\\_Solar\\_Photovoltaic\\_Array\\_Location\\_and\\_Extent\\_Data\\_Set\\_for\\_Remote\\_Sensing\\_Object\\_Identification/3255643](https://figshare.com/collections/Full_Collection_Distributed_Solar_Photovoltaic_Array_Location_and_Extent_Data_Set_for_Remote_Sensing_Object_Identification/3255643).



*OpenCV*. URL: <https://opencv.org/>.