NRG4CAST

ENERGY FORECASTING

Big Data Techniques For Supporting Accurate Predictions of Energy Production From Renewable Sources

Michelangelo Ceci, Donato Malerba, Giuseppe Manco, Elio Masciari, Aleksandra Rashkovska

ICAR-CNR, UNIBA

SEVENTH FRAMEWORK PROGRAMME

I know what you are thinking…

NRG4CAST
ENERGY FORECASTING

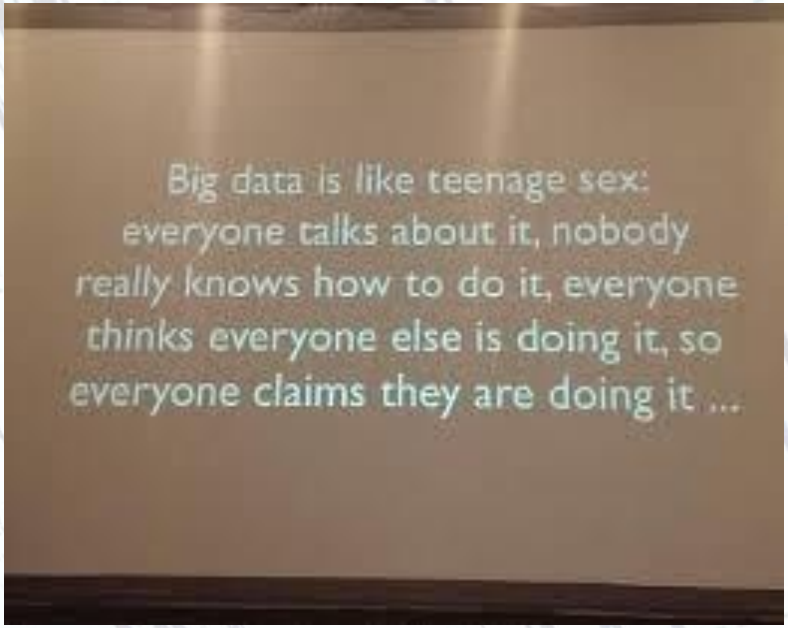
Big Data (in the mass culture)

Big Data (a pessimistic vision)

- Large volumes, Large diversification, High Speed:
  - 3V initial paradigm
    - Volume
    - Velocity
    - Variety

Big Data (an optimistic vision)

NRG4CAST ENERGY FORECASTING



- Add more V:
  - Veracity
  - Variability

Big Data (for real life)

NRG4CAST ENERGY FORECASTING

"Let's say you want to save millions of dollars — you just push this button here..."

The last V:

- Value

How to turn data into value (Reality)

- Data Mining
- Querying
- Exploratory Analysis

Renewable energy:
a strategic sector for all European countries

It contributes to reducing energy dependency from foreign countries and pollution emission.

In this sector there is **abundance of data** (generated by production plants) to organize, model and analyze in order to create value.

Issues:

- Data produced continuously at high rate
- Lack of scalability of the underlying algorithms for storing/analyzing these data
- Complexity of the data (heterogeneity)
- Presentation of results and interpretation by non-technical domain experts

The data mining task

Energy sector has its own market organization regulated by demand / offer rules which define the hourly / daily price of the energy.

Each single power source may influence the final clearing price.

Thus, it is important

- to monitor the local / global production and consumption of energy
- store historical data,
- design new and reliable **prediction models**.

The data mining task

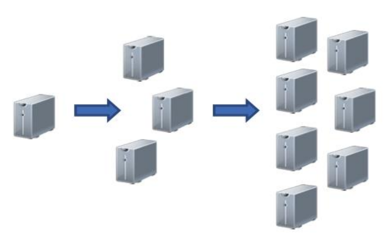
Specific goal: develop a method to predict one day ahead production (hour by hour) of heterogeneous sources (photovoltaic, wind, biomass, etc...)

Data:

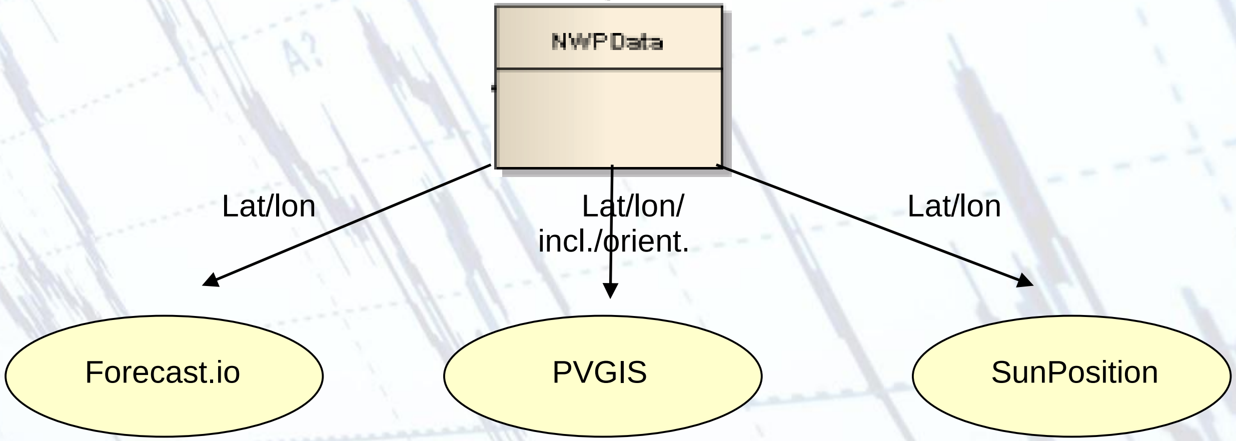
- Historical data and real time data of production, continuously produced at regular time intervals by sensors placed on each plant of interest
- Weather predictions gathered from *NWP* *(Numerical Weather Prediction)* models
- Irradiance predictions

uncontrollable factors

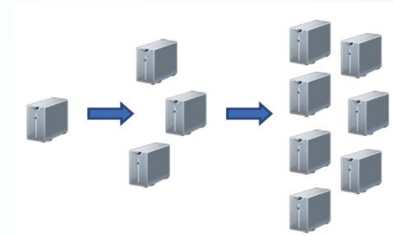
Data collection and loading



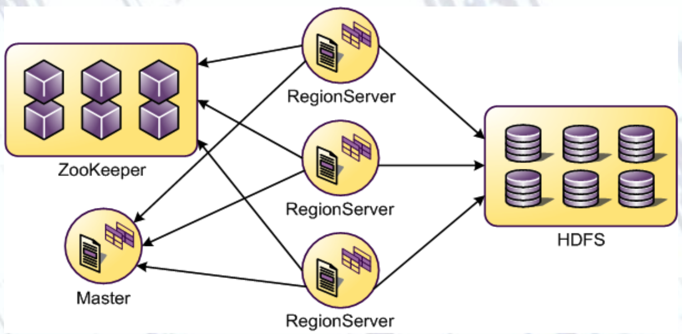
- Production data are collected from **sensors** placed on plants
- One-day-ahead weather data are collected from **Forecast.io**
- Irradiance data are collected from **PVGIS**
- Altitude and azimuth of the sun are collected from **SunPosition**



Data collection and loading

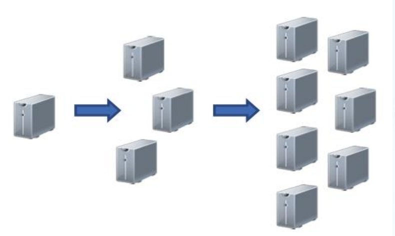


- Data gathered from sensors and NWP models will be stored in HBASE on HDFS



- The distributed approach allows to compute the analysis task on a cluster of nodes, leading to an improved efficiency, higher scalability and availability. This task is accomplished relying to Hadoop framework and Mapreduce

Logical design in HBASE



Four tables:

- **Plants**

  Stores plant data such as the coordinates and the maintenance operations

- **Measure**

  Stores all data observed by sensors placed on plants

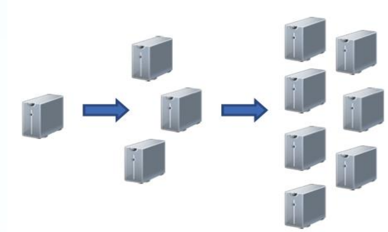
- **Predicted**

  Stores the measures predicted by mining algorithms

- **Weather Data**

  Stores weather data collected from external sources

Logical design in HBASE



Plants

row key: concat (plantType + plantID)

family_info

Lat, Long, altitude*, max_power, model, vendor, rated_power, surface*, total_surface*, inclination*

family_maintenance

row key timestamp

description

Measures

row key: concat (plantID + reverse timestamp + measurement type)

external_temp
cell_temp
Irradiance

windspeed
winddirection

outputpower

Predicted Measures

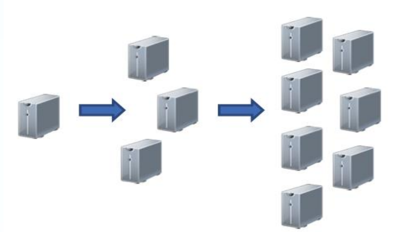
row key: concat (plantID + reverse timestamp + measurement type)

Prediction by algorithm

row key prediction_algorithm

outputpower

Logical design in HBASE



Weather Data

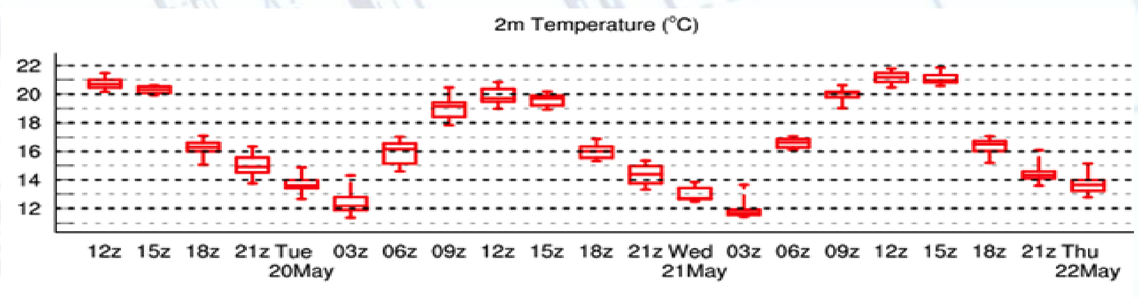
row key: concat (geoHash + reverse timestamp + measurementType + servID)

family_collected	family_predicted
row key: server	**row key: server**
temperature	temperature
cloud_cover	cloud_cover
wind_speed	wind_speed
wind_direction	wind_direction
pressure	pressure
humidity	humidity
precipitations	precipitations
global_irradiance,	global_irradiance,
direct_irradiance	direct_irradiance
clear_sky_direct_irradiance	clear_sky_direct_irradiance
diffuse_irradiace	diffuse_irradiace
clear_sky_2axes_direct_irradiance	clear_sky_2axes_direct_irradiance
2axes_diffuse_irradiance	2axes_diffuse_irradiance
2axes_global_irradiance	2axes_global_irradiance
clear_sky_normal_direct_irradiance	clear_sky_normal_direct_irradiance

Data preparation

- Noisy and missing data (due to faults on sensors) are corrected thanks to historical measurements stored on other databases

TIMESTAMP	TOTAL	PINV1	PINV2	CINV1	CINV2	TINV1	TINV2	TEMP	IRR	KWH
101212 09.00	788	398	390	735	726	553	551	21	637	2107266
101212 10.00	777	392	385	719	718	556	550	-30	?	2107462
101212 11.00	762	384	378	706	702	559	552	21	650	2107653



Data preparation

- Sensors located on plants can be covered by ostacles or dirt.
- Hence, irradiance measured locally is often lower than irradiance extracted by NWP models.
- Training a model using sensors data and using it for predictions with NWP data can lead to inaccurate predictions.

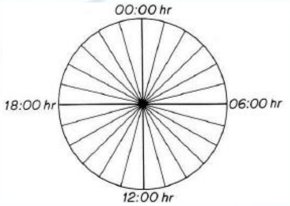
Solution:

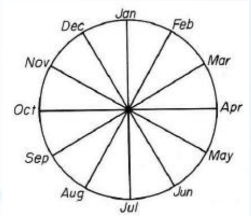
- Calculate the **percentage of change** between monthly NWP irradiance and irradiance detected by sensors on historical data (same month at the same hour), to understand how much they differ;

- Alter, future NWP data are normalized accordingly.

$$Pc = \frac{y - x}{x} * 100$$

Temporal autocorrelation





- **Issue**: Weather data is inherently seasonal / cyclical.

e.g: summer days are featured by an increased irradiance compared to winter days.

- Hence, days that are closer to the prediction day should be taken more in consideration by the model, in order to make a more reliable prediction.

Idea:

- Usage of **directional statistics** to calculate the distance ***dist*** between the prediction day (e.g.: tomorrow) and days used in to train the model (historical data)
- Transform distance into similarity ***(1-dist)***
- Incorporate such similarity as a feature (independent variable) in the model

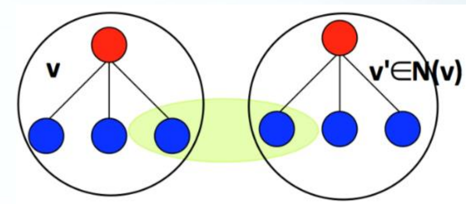
$$dist(d1,d2) = \min(mod_{366}|d1-d2|),(mod_{366}|-d1+d2+366|)$$

E.g: Prediction for: 25th December (day 360)

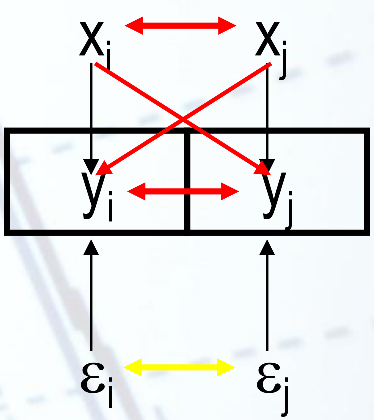
dist(360, 5) = 0.03 5th January is closer than 3rd May

dist(360, 122) = 0.34

Spatial autocorrelation



Issue: The proximity of sensors induces spatial autocorrelation in data: this violates the assumption of instances being independent and equally distributed. Usage of statistical techniques to handle spatial autocorrelation between plants



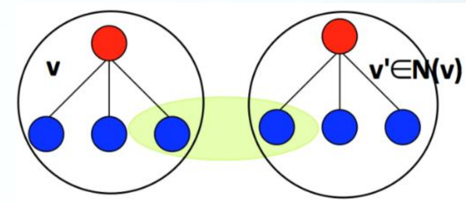
•First and very simple solution: include **Latitude and Longitude** of the plants (not new: already applied in [Stojanova et al. 2012])

•**PCNM technique**

Given a matrix of distances between plants, calculate eigenvectors that maximize **Moran's I** statistic and incorporate them in the predictive model;

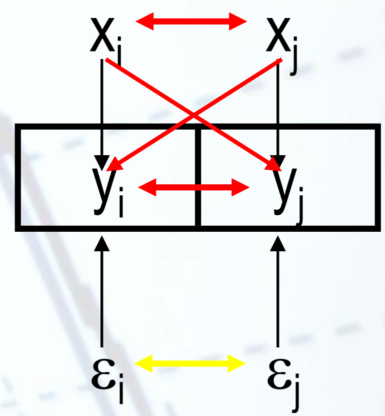
$$I = \frac{N}{\sum_i \sum_j w_{ij}} \frac{\sum_i \sum_j w_{ij}(X_i - \bar{X})(X_j - \bar{X})}{\sum_i (X_i - \bar{X})^2}$$

Spatial autocorrelation



- **LISA technique**

  Given a neighborhood matrix of plants, calculate local **Moran's I** statistic for each feature on each plant at each timestamp and and incorporate them in the predictive model;



$$w_{i,j} = 1/|Ni|$$

For each plant (matrix row)
    For each neighbor (matrix column > 0)
        For each timestamp
            For each feature (temperature, humidity, ...)

$$z_i = \frac{x_i - \bar{x}}{SD_x}$$

$$I_i = z_i \sum_j w_{ij} z_j$$

One final I for each timestamp, plant, feature

The learning phase

- Resilient Propagation (**RPROP+**) is one of the best general purpose neural network training methods which use backpropagation
- It performs a direct adaption of weight step based on local gradient information

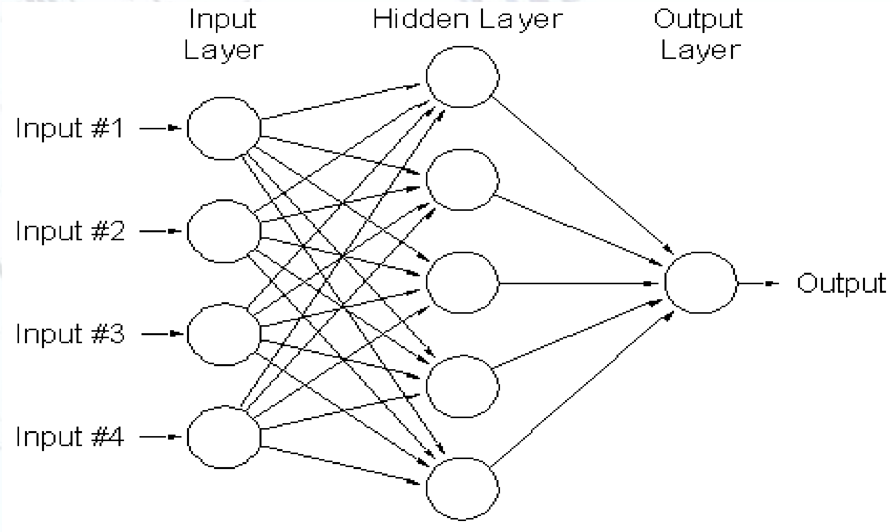
- Basic principle of **RPROP+** is to eliminate the harmful influence of the size of the partial derivative on the weight step;
- It considers only the sign of the derivative to indicate the direction of the weight update.

Already used for renewable energy prediction in [Bessa et al. 2009] where, instead of the MSE criterion, MEE, MCC and MEEF criteria are used (Parzen window method).

The learning phase

NRG4CAST ENERGY FORECASTING

Two alternatives: Method 1 (Hourly)



Day expressed as **24 data rows**

Hr1, f1, ..., fn, KWH

...

Hr24, f1, ... , fn, KWH

Output: prediction for a specific hour of the next day (possibly taking in account previous predictions)

The learning phase

Two alternatives: Method 2 (Daily)

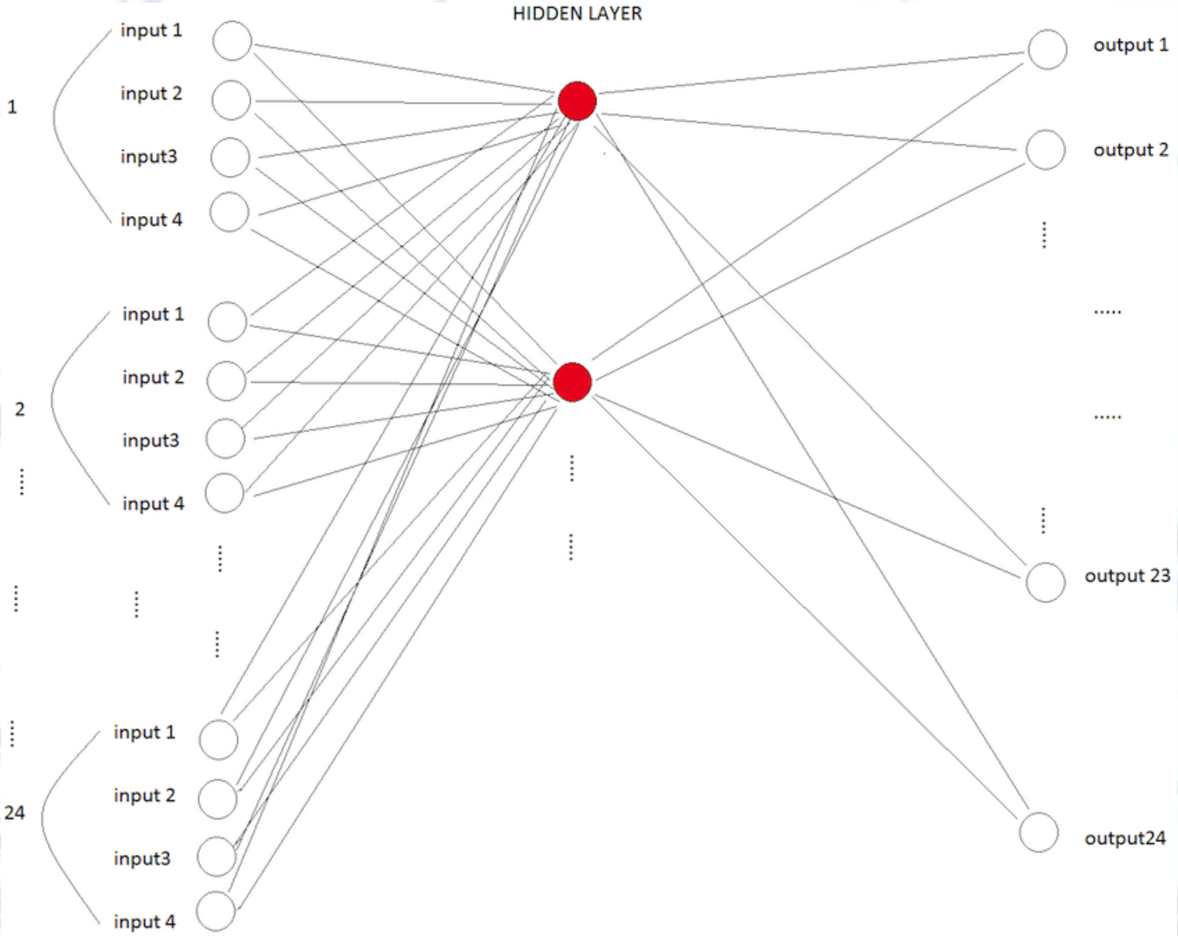
Considers possible dependence among hours

Day expressed as a **single data row**

Hr1, f1, ... , fn,KWH , ... , Hr24, f1, ... ,fn , KWH

Output: prediction of a 24-elements vector (**structured output prediction**)

The learning phase



Experimental settings

NRG4CAST
ENERGY FORECASTING

- **Hourly prediction:**

idplant, idbrand, lat, lon, day, daySim, hour, temperature, irradiance, pressure, windspeed, humidity, icon, dewpoint, windbearing, cloudcover, temperatureI, irradianceI, pressureI, windspeedI, humidityI, dewpointI, windbearingI, cloudcoverI, pcnm1, pcnm2,,.., pcnmN, **kwh**

- **Daily prediction:**

idplant, idbrand, lat, lon, day, daySim hour2, temperature2, irradiance2, pressure2, windspeed2, humidity2, icon2, dewpoint2, windbearing2, cloudcover2, temperatureI2, irradianceI2, pressureI2, windspeedI2, humidityI2, dewpointI2, windbearingI2, cloudcoverI2, … , hour20, temperature20, irradiance20, pressure20, windspeed20, humidity20, icon20, dewpoint20, windbearing20, cloudcover20, temperatureI20, irradianceI20, pressureI20, windspeedI20, humidityI20, dewpointI20, windbearingI20, cloudcoverI20, pcnm1, pcnm2,,.., pcnmN, **kwh2, kwh3, kwh4, kwh5, kwh6, kwh7, kwh8, kwh9, kwh10, kwh11, kwh12, kwh13, kwh14, kwh15, kwh16, kwh17, kwh18, kwh19, kwh20**

Space	Time
No spatial	No temporal
Lat Lon	No cyclic
LISA	Cyclic
PCNM	

Predicted variables

4 * 3 * 2 = 24 settings

|Space| * |Time| * |Hourly,Daily|

Experimental results

Data collected by SunElectrics over the time period between 2012 and 2014

17 photovoltaic plants (in Italy)

Data collected every 15 minutes

Training data:	2012-2013 (731 days)
Testing data (backpropagation):	2014 (126 days)

Experimental results

NRG4CAST ENERGY FORECASTING

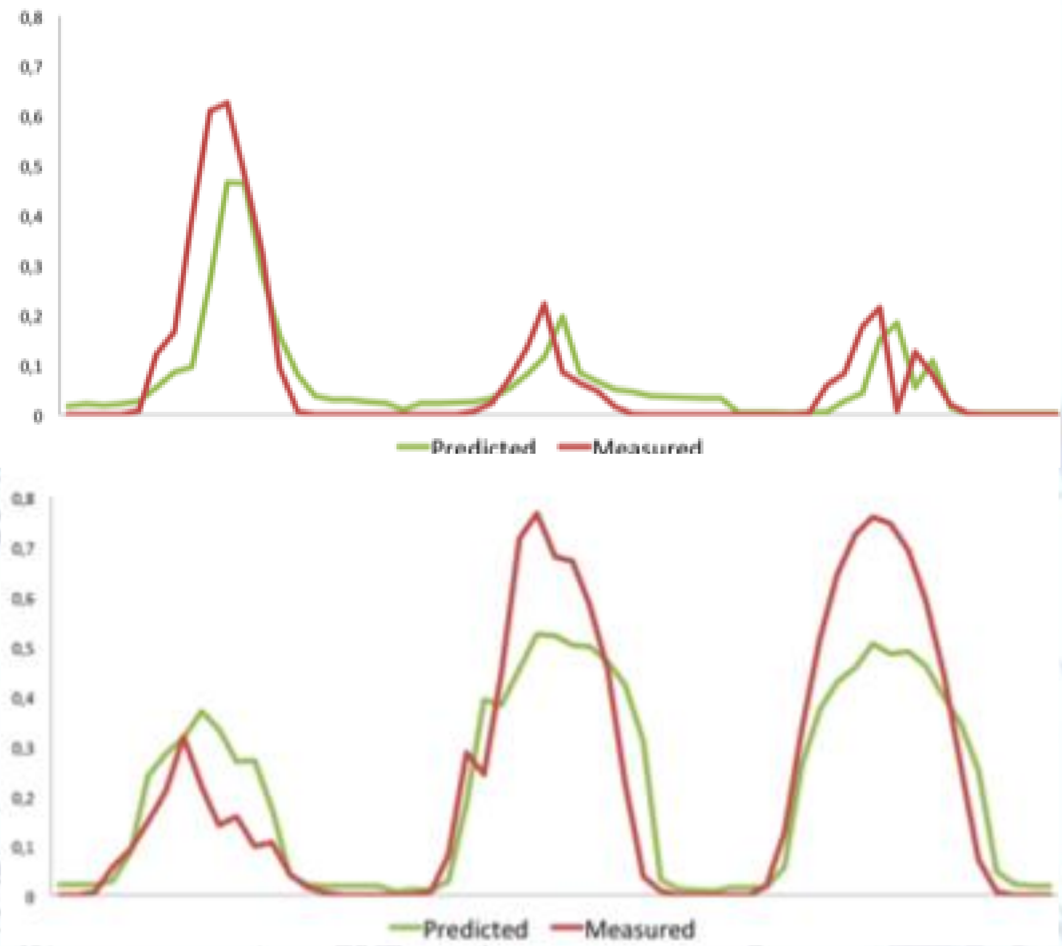
	Hourly								
	No Temporal			**Non Cyclic**			**Cyclic**		
	RMSE	MAE	% Impr.	RMSE	MAE	% Impr.	RMSE	MAE	% Impr.
No Spatial	0,121	0,080	16,622	0,120	0,079	17,410	0,121	0,083	16,585
Lat Lon	0,119	0,078	18,254	0,120	0,078	17,443	0,121	0,083	16,674
LISA	0,117	0,076	**19,617**	0,118	0,077	18,955	0,117	0,077	19,510
PCNM	0,120	0,079	17,254	0,123	0,080	15,796	0,124	0,081	15,003

	Daily								
	No Temporal			**Non Cyclic**			**Cyclic**		
	RMSE	MAE	% Impr.	RMSE	MAE	% Imp	RMSE	MAE	% Imp
No Spatial	0,111	0,068	23,966	0,109	0,068	24,810	0,108	0,066	26,095
Lat Lon	0,109	0,067	25,369	0,111	0,069	23,915	0,106	0,065	**27,401**
LISA	0,109	0,067	24,858	0,110	0,067	24,594	0,107	0,066	26,760
PCNM	0,109	0,068	24,889	0,109	0,067	25,445	0,107	0,066	26,521

Persistence	
RMSE	MAE
0,146	0,085

Experimental results

NRG4CAST ENERGY FORECASTING



Conclusions

- Spatial and temporal features help to achieve a better prediction
- Prediction is better in case of structured outputs (daily settings), probably because of the implicit consideration of the dependance of the predictions at consecutive hours

Ongoing work:

- Incorporate autocorrelation measures in the update rule of the algorithm
- Consider other predictive approaches
- Evaluate the system on other datasets

Questions?