http://dx.doi.org/10.1109/TII.2016.2604758

Predictive Modeling of PV Energy Production: How to Set Up the Learning Task for a Better Prediction?

Michelangelo Ceci, Roberto Corizzo, Fabio Fumarola, Donato Malerba, *Member, IEEE,* and Aleksandra Rashkovska, *Member, IEEE*

***Abstract*—In this paper, we tackle the problem of power prediction of several photovoltaic (PV) plants spread over an extended geographic area and connected to a power grid. The paper is intended to be a comprehensive study of one-day ahead forecast of PV energy production along several dimensions of analysis: *i)* The consideration of the spatio-temporal autocorrelation, which characterizes geophysical phenomena, to obtain more accurate predictions. *ii)* The learning setting to be considered, i.e. using simple output prediction for each hour or structured output prediction for each day. *iii)* The learning algorithms: We compare artificial neural networks, most often used for PV prediction forecast, and regression trees for learning adaptive models. The results obtained on two PV power plant datasets show that: taking into account spatio/temporal autocorrelation is beneficial; the structured output prediction setting significantly outperforms the non-structured output prediction setting; and regression trees provide better models than artificial neural networks.**

***Index Terms*—PV energy prediction, spatial and temporal autocorrelation, structured output, regression trees, ANNs.**

I. Introduction

THE urgent need to reduce pollution emission has made renewable energy a strategic European Union (EU) and international sector. This has resulted in an increasing presence of renewable energy sources and thus, significant distributed power generation. The main challenges faced by this new energy market are grid integration, load balancing and energy trading. First, integrating such distributed and renewable power sources into the power grid, while avoiding decreased reliance and distribution losses, is a demanding task for the smart grid effort. In fact, renewable power sources, such as photovoltaic array, are variable and intermittent in their energy output, because the energy produced may also depend on uncontrollable factors, such as weather conditions. Second, the main players in the energy market – the distributors and smaller companies that act between offer (traders) and request in the supply chain – have to face uncertainty not only in the request but also in the offer, when planning the energy supply for their customers. Third, the power produced by each single source contributes in defining the final clearing price in the daily or hourly market [4], thus making the energy market very competitive and a true maze for outsiders.

In order to face these challenges, it is of paramount importance to monitor the production and consumption of energy, both at the local and global level, to store historical data and to design new, reliable prediction tools. In this work, we focus our attention on photovoltaic (PV) power plants, due to their wide distribution in Europe. During the last years, the forecast of PV energy production has received significant attention since photovoltaics are becoming a major source of renewable energy for the world. Forecast may apply to a single renewable power generation system [20], or refer to an aggregation of large numbers of systems spread over an extended geographic area [3][19]. Accordingly, different forecasting methods are used. Furthermore, forecasting methods also depend on the tools and information available. Diverse resources are used to generate solar and PV forecasts depending on the forecast horizon considered, ranging from measured weather and PV system data to satellite and sky imagery cloud observations used for very short-term forecasts (0 to 6 hours ahead), to Numerical Weather Prediction (NWP) models used for horizons beyond approximately six hours [15]. Several works clarify that the best approaches make use of both measured data and NWP [18][19].

In the literature, several data mining approaches have been proposed for renewable energy power forecasting. Researchers typically distinguish between two classes of approaches: physical and statistical. The former relies on the refinement of NWP forecasts with physical considerations (e.g. obstacles and orography) [5] or measured data (approach often referred to as Model Output Statistics or MOS) [18][19], while the latter is based on models that establish a relationship between historical values and forecasted variables. Statistical approaches may or may not take into account NWP data. Some of them are based on time series [8], while others learn adaptive models from data, like autoregressive (AR) models [3], artificial neural networks (ANNs) [20], or SVM classifiers [21]. In this respect, it has been noted that physical property behavior (e.g. wind speed and solar irradiation) exhibits a trail called concept drift, i.e., they change characteristics over time [4]: Adaptive models are generally considered to produce more reliable

Manuscript received April 20, 2016. Revised July 15, 2016. Accepted for publication August 15, 2016. This work was supported by the projects ”Vi-POC: Virtual Power Operation Center” (PAC02L1_00269), funded by the Italian Ministry of University and Research, and ”MAESTRA - Learning from Massive, Incompletely annotated, and Structured Data” (Grant number ICT-2013-612944), funded by the European Commission. The computational work has been carried out on the resources provided by the projects ReCaS (PONa3_00052) and PRISMA (PON04a2_A). The authors also wish to thank Lynn Rudd for her help in reading the manuscript.

M. Ceci, R. Corizzo, F. Fumarola, D. Malerba and A. Rashkovska are with the Department of Computer Science, University of Bari Aldo Moro, via Orabona 4, 70125 - Bari, Italy (e-mail: michelangelo.ceci@uniba.it; roberto.corizzo@uniba.it; fabio.fumarola@uniba.it; donato.malerba@uniba.it). A. Rashkovska is also with the Department of Communication Systems, Jožef Stefan Institute, Jamova 39, 1000 - Ljubljana, Slovenia (e-mail: aleksandra.rashkovska@ijs.si).

predictions regarding concept drift, but require a continuous training phase. Combinations of statistical (ANN and SVM) and physical (MOS) approaches for renewable energy power forecasting have also been recently investigated [7].

Despite the existence of such data mining algorithms applied in renewable energy power forecasting for learning adaptive models [3][4][20][21], there is no consensus about the spatio-temporal information to be taken into account, the learning setting to be considered and the learning algorithms to be used. This paper considers all these aspects as dimensions of analysis and investigates their real contribution in renewable energy power forecasting. The paper is structured as follows. Section II gives the motivation and contribution of the study. Section III formalizes the problem, how spatial and temporal autocorrelation components are considered and how the (non-)structured output learning task is defined. Section IV presents the data collection and preprocessing. The dimensions of analysis stated above are addressed in Section V, where experiments are reported and results are discussed. Finally, Section VI concludes the paper.

II. Motivation and Contribution

The motivation for this paper comes from the different learning settings that can be found in the literature for the task at hand. Concerning the *spatio-temporal information* to be taken into account, it is noteworthy that most of the previously referenced work consider forecasting solutions for single plants and ignore the information collected from/at other plants/sites in the vicinity, even when this would be easily accessible (through spatial information). In this work, we show that this information loss may result in reduced accuracy of the forecasting models, and we advocate an approach for learning forecasting models from data related to multiple plants. Differently from the few works in the literature that consider multiple plants [3][19], our analysis also leverages the spatio-temporal autocorrelation that characterizes geophysical phenomena, such as weather conditions, to make more accurate predictions.

Indeed, site proximity of PV plants introduces *spatial* autocorrelation[1] in functional annotations and leads to the violation of the usual assumption that observations are independently and identically distributed (i.i.d.). Although the explicit consideration of these spatial dependencies brings additional complexity to the learning process, it generally leads to increased accuracy of learned models [22].

In addition, the production of PV plants is also affected by *temporal* autocorrelation, since it: *i)* tends to have similar values at a given time in close days, *ii)* has a cyclic and seasonal (over days and years) behavior, *iii)* tends to show the same trend over time. While adaptive approaches (which typically employ stream mining algorithms) deal with *i)* and *iii)*, they may fail to consider *ii)*, since they tend to better represent the most recently observed concepts, forgetting previously learned ones [12]. On the contrary, time series-based approaches are able to deal with *iii)*, but may fail to consider *i)* and *ii)*. In fact, they typically require the size of the temporal horizon as input: Considering a short-term horizon (e.g. daily) excludes a long-term horizon (e.g. seasonal) and vice versa.

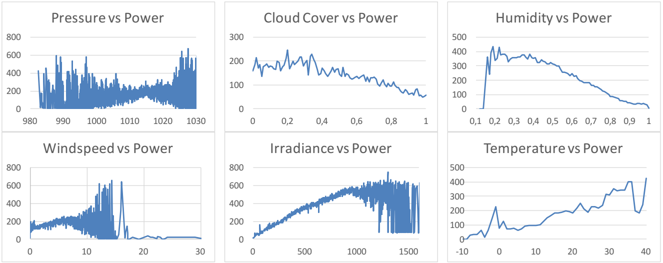


Fig. 1. Trend of pressure (hPa), cloud cover (%), humidity (%), wind speed (m/s), irradiance (W/m2), temperature (°C) (X-axis) with respect to the energy produced (KWh) (Y-axis).

Although several approaches combine both the spatial and the temporal dimensions [10] from stream data, they have not been applied in the context of renewable energy power prediction. Moreover, they do not take the spatial autocorrelation phenomenon into account and, thus, do not exploit the spatial structure of the data [6]. In this paper, similarly to other approaches, we exploit NWP to benefit from uncontrollable factors, but additionally, we evaluate at which extent taking into account spatial and temporal autocorrelation is beneficial. While spatial autocorrelation is taken into account by resorting to two well known techniques in spatial statistics, temporal autocorrelation is considered by resorting to directional statistics.

Concerning the *learning setting*, in the existing work, the classical solution is to learn a predictive model which predicts the energy that will be produced at a specific hour and day in the future. This approach, however, appears to be too limiting if we consider that two consecutive hours are seen as independent examples. An alternative solution is to resort to approaches that learn predictive models for structured output (structured output prediction) [16]. In principle, this approach should be able to catch (and model) dependence between consecutive hours of the same day, just as some time-series methods do for short-term prediction.

Concerning the *data mining algorithm*, we investigate the predictive performance obtained with two different algorithms, one which learns ANNs and another one which learns regression trees. While ANNs have been extensively used for energy prediction [4], regression trees have received significantly less attention. Both algorithms are also able to deal with the non-linearity of production with respect to some of the considered features (see Fig. 1). In any case, we study and compare their performance for the problem at hand. Actually, in our framework, we can plug-in any learner which can learn models for both non-structured and structured output.

The contributions of the paper aim to provide a comprehensive analysis of the problems described before and to understand what matters if we want to achieve good and reliable predictions. In particular, the contributions include: *1)* The explicit consideration, with two different solutions, of the spatial autocorrelation and the investigation of its effect at different extents in predictive modeling of energy production. In this way, weather conditions (e.g. temperature,

[1] Correlation among data values which is strictly due to the relative spatial proximity of the objects that the data refer to.

solar radiation, wind direction and (wind) speed, quantity of rain) are appropriately exploited for forecasting purposes. *2)* The explicit consideration of the temporal autocorrelation and the investigation of its effect at different extents in predictive modeling of energy production, in order to deal with non-stationary (cyclic and seasonal) data, in an adaptive model. *3)* An investigation of the effect of the structured output prediction approach. *4)* An investigation of the effect of the specific learning algorithm used. *5)* The identification, by means of a feature selection step, of the best variables to be used for prediction. We orthogonally investigate all the aspects discussed before through an extensive experimental evaluation on two datasets for PV power plants in Italy and USA.

III. Method

The task we intend to perform is to predict PV power generation from *i)* historical data on power production, *ii)* weather forecast data provided by NWP systems, *iii)* weather information collected by sensors, *iv)* geographic coordinates of the plants, *v)* additional features representing spatial and temporal autocorrelation. Similarly to [8], the output is a fine-grained prediction for the next day at one hour intervals.

The learning algorithms update the prediction models every day. We use historical weather information collected by sensors as features in the training phase, whereas we use weather forecast data provided by NWP systems as features for predictions. Formally, let P_i be the i-th plant, we define:

- $\pi_i =< x, y >$, the geographic coordinates of P_i,
- $\alpha_{i,j,h}$, a vector representing the sun's position at the location of P_i, on day j and at hour h,
- $p_{i,j,h}$, a vector representing properties of P_i,
- $w_{i,j,h}$, a vector for the observed weather data of P_i,
- $w'_{i,j,h}$ a vector that represents NWP forecast data for P_i,
- $s_{i,j,h}$, a vector modeling spatial autocorrelation at P_i,
- $t_{i,j,h}$, a vector modeling temporal autocorrelation at P_i,
- $y_{i,j,h}$, a value representing the production (KWh) of P_i.

In our approach, we use π_i, $\alpha_{i,j,h}$, $p_{i,j,h}$, $w_{i,j,h}$, $s_{i,j,h}$, $t_{i,j,h}$ (independent input features) and $y_{i,j,h}$ (dependent/target variable), for training purposes and π_i, $\alpha_{i,j,h}$, $p_{i,j,h}$, $w'_{i,j,h}$, $s_{i,j,h}$ and $t_{i,j,h}$ for prediction purposes. The value of the predicted attribute – the energy production – is available only during training. It is noteworthy that $p_{i,j,h}$ is used to represent data about the plants, which are valid at a certain time-point. Examples of features used in $p_{i,j,h}$ are: age of the plant, number of working inverters and maximum plant production. These data allow the learning algorithm to directly predict the production, instead of the percentage of the production with respect to the maximum plant production.

The applied spatial and temporal autocorrelation techniques are discussed in the next two subsections. The last subsection describes the two learning settings for predicting $y_{i,j,h}$, namely structured and non-structured output prediction.

A. Spatial Autocorrelation

The proximity of sensors induces spatial autocorrelation in the data. According to [17], the inappropriate treatment of sample data with spatial dependence could obfuscate important insights and observed patterns may even be inverted when spatial autocorrelation is ignored. Taking autocorrelation into account allows the models to avoid overfitting and exploit information coming from close sites. To accommodate several forms of spatial correlation, various models have been developed in the field of spatial statistics. The most known types are the spatial lag model and the spatial cross-regressive model [2]. While the former considers autocorrelation on the target variables, the latter considers cross-correlation between input features at one site and target variables at other sites, as well as cross-correlation between input features at one site and input features at other sites (see Fig. 2).

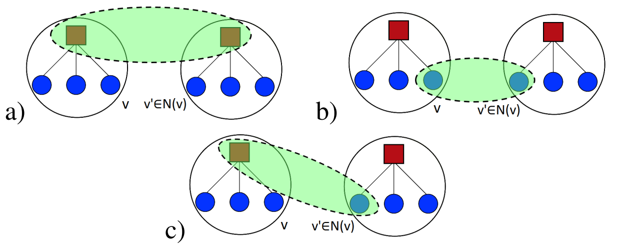


Fig. 2. Different types of autocorrelation (dashed lines): Squares and circles represent target and input features, respectively; bigger circles represent different sites; $N(v)$ represents the neighborhood of the site v. (a): Spatial lag model. (b) and (c): Two forms of spatial cross-regressive model.

We use two spatial statistics when building the cross-regressive model between PV plants: 1) the Local Indicator of Spatial Association (LISA) for representing a local measure of spatial autocorrelation [1]; 2) the Principal Coordinates of Neighbor Matrices (PCNM) for representing the spatial structure in the data [9]. Compared to the classical STAR model [23], which combines spatial and temporal information in an autoregressive model, the advantage of the solution we adopt is that we are able to embed spatial and temporal information in new features rather than in the model (but still taking autocorrelation into account). This gives us the opportunity to plug-in any off-the-shelf learning algorithms and to separately investigate the contribution of spatial and temporal information.

To compute the LISA, the spatial neighborhood of an entity (i.e. PV plant) is expressed as a matrix, and for each spatial entity and data point observed in a time horizon, the local Moran's I index [1] is computed using the matrix. More precisely, given n PV plants, the first step is to define a neighborhood matrix δ of size $n \times n$, such that:

$$\delta[P_a, P_b] = \begin{cases} 1 & \text{if } dist(P_a, P_b) < maxDist \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where P_a and P_b are two of the n plants. The $maxDist$ threshold is a user-defined parameter that determines the effect of the autocorrelation. The elements of δ are then transformed: $\delta'[P_a, P_b] = \frac{1}{|N(P_a)|}\delta[P_a, P_b]$, where $N(P_a)$ is the set of nodes which are at a distance less than or equal to $maxDist$ with respect to P_a. In this way, the sum of the elements on each row in δ' is either 0 or 1.

The subsequent step consists in computing the deviation of the variable of interest with respect to the mean according to the z-score normalization [14]. Since in our approach we are interested in identifying the contribution of the neighborhood for each feature, this computation is performed for each feature considered (e.g.: temperature, humidity, etc.). More formally, for a plant P_a at day j and hour h, we compute:

$$z^{(x)}_{a,j,h} = \frac{x_{i,j,h} - \overline{x}}{\sigma_x}, \qquad (2)$$

where x is a generic variable used for representing either weather condition, weather forecast or plant information (that is, a generic element of the vectors w, w' and p), $\overline{x}$ represents the average of the variable x and σ_x represents the standard deviation of x. On the basis of $z^{(x)}_{a,j,h}$, it is possible to compute the local Moran's I for the variable x of the plant P_a for day j and hour h (according to [1]): $I^{(x)}_{a,j,h} = z^{(x)}_{a,j,h} \cdot \sum_{P_i \in N(P_a)} \delta'[P_a, P_i] \cdot z^{(x)}_{i,j,h}$.

According to [25], we incorporate local spatial autocorrelation terms as predictors in regression equations (in the vector $s_{a,j,h}$, which has one element for each possible variable x), giving place to autoregressive models.

The PCNM is used to define new features which represent the spatial structure of the data, so to exploit autocorrelation in the learning phase. Its computation has three steps:

1) The Euclidean (geographic) distance matrix D between plants is computed ($D = [d_{i,j}] = [dist(P_i, P_j)]$).
2) A threshold value t is chosen to construct a truncated distance matrix $D^* = [d^*_{ij}]$ as follows:

$$d^*_{ij} = \begin{cases} d_{ij} & \text{if } d_{ij} \leq t \\ 4t & \text{otherwise.} \end{cases}$$

3) A principal coordinate analysis (PCoA) of the truncated distance matrix D^* is performed. This analysis consists in the diagonalization of $\varDelta$, where:

$$\varDelta = -\frac{1}{2}\left(\mathbf{I} - \frac{1 \cdot 1^t}{n}\right)\mathbf{D_2^*}\left(\mathbf{I} - \frac{1 \cdot 1^t}{n}\right) \qquad (3)$$

with $\mathbf{D_2^*} = [(d^*_{ij})^2]$, $\mathbf{I}$ be the identity matrix and 1 be a vector of 1s.

It has been proven (see [9]) that the eigenvectors of $\varDelta$ are vectors with unit norm maximizing Moran's I under the constraint of orthogonality, whereas the eigenvalues of this matrix are equal to Moran's I coefficients of spatial autocorrelation (post-multiplied by a constant). They can also be either positive or negative (because the original Euclidean distance matrix has been truncated). Eigenvectors associated with high positive (or negative) eigenvalues have high positive (or negative) autocorrelation and describe global and local structures. Since we are interested in considering only positive spatial autocorrelation, only eigenvectors corresponding to positive eigenvalues are kept and used as spatial descriptors.

The principal coordinates of each spatial descriptor (used as features in $s_{i,j,h}$) are obtained by scaling each eigenvector u_k of $\varDelta$ to the length $\sqrt{\lambda_k}$, where λ_k is the eigenvalue associated with eigenvector u_k. Finally, the value of t used in D^* is defined as the maximum distance between two PV plants ($t = \max\limits_{i,j} d_{i,j}$), in order to guarantee that the data are all connected.

B. Temporal Autocorrelation

Weather data is inherently seasonal/cyclical. For instance, summer days are featured by an increased irradiance compared to winter days. Moreover, if we consider an absolute time reference, the irradiance is almost equal for two (close) days at the same hour. Hence, we expect to increase the reliability of predictions if days closer to the prediction (target) day are more influential in the model.

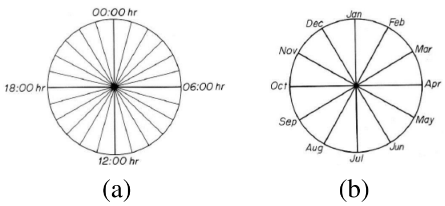


Fig. 3. (a) Daily circumference. (b) Yearly circumference.

To account for both seasonal and daily cyclicity, we use two alternative solutions. The first solution is simple and consists in considering, as input features (independent variables), the values of time and day scaled in the range [0,1] for both training and target days. The second solution is to resort to directional statistics which represent directional or circular distributions of the data by "wrapping" the probability density function around the circumference of a circle of unit radius. We follow this approach by "wrapping" information collected at a specific time-point around two circumferences, namely the daily circumference and the yearly circumference (see Fig. 3). The former catches the distribution of the production over the different hours of the day, while the latter catches the distribution of the production over the 365 days of the year.

More specifically, for each day in the training (historical) data, we compute its radial distance d_r on the circumference from the target day. Since the learning algorithm is able to consider the similarity between two values, the value $2\pi - d_r$ is incorporated as input feature in the model (in $t_{i,j,h}$). Note that this approach requires an update of the model every day, which is in line with the learning setting we consider. However, the same approach cannot be adopted to catch the distribution of the production over the different hours of the day since this would require updating the model every hour. The solution we adopt in this case is to directly include in the model the radial value of the specific hour h_r as input feature (independent variable in $t_{i,j,h}$). The disadvantage is that we can not catch similarities in the energy produced between the last and the first hours of the day when, however, PV energy is not produced.

C. The learning setting: Structured and non-structured

As previously mentioned, we also investigate the application of structured output prediction models to the specific task. In particular, we consider multi-target models where, instead of a single model that predicts the production at a single hour, we have a single model that predicts 24 output variables at the same time. In principle, learning multi-target models should present some benefits over learning a local model for predicting the production at each hour. Indeed, it is generally recognized that structured models are typically easier to interpret, perform better and overfit less than single-target predictions [16]. In this specific application, multi-

TABLE I
CORRESPONDENCES BETWEEN INDEPENDENT VARIABLES IN THE NON-STRUCTURED AND STRUCTURED OUTPUT SETTINGS.

Non-structured	Structured	
$\alpha_{i,j,h}$	$\alpha_{i,j,h=1:24}$	
$p_{i,j,h}$	$p_{i,j,h=1:24}$	
$w_{i,j,h}$	$w_{i,j,h=1:24}$	
$w'_{i,j,h}$	$w'_{i,j,h=1:24}$	
$s_{i,j,h}$	$s_{i,j,h=1:24}$	
$t_{i,j,h}$	$t_{i,j}$	(only daily information)

target models can also exploit the dependencies between the productions at two different hours of the same day. The main issue is the collinearity problem, since the linear dependence between attributes may negatively affect the learned model.

Formally, when a non-structured output prediction setting is used, the predicted value is $y_{i,j,h}$ (for the plant i, at the day j and hour h). On the contrary, if a structured output prediction setting is used, the predicted value is a vector $[y_{i,j,1}, y_{i,j,2}, \dots, y_{i,j,24}]$ (for the plant i, at the day j). Since in the structured output prediction setting, the unit of analysis is the day and in the non-structured output prediction setting the unit of analysis is the hour, the independent variables are different. In fact, in the non-structured output setting, each training instance represents a single hour, while in the structured output approach each training instance represents a single day. This means that, in order to keep the same information between the two settings, it is necessary to reserve 24 variables in the structured output setting to represent the value observed for the same property. Formally, the correspondences defined in Table I hold.

IV. DATA COLLECTION AND PREPROCESSING

Coherently with the problem definition provided in Section III, we distinguish between data locally collected by sensors installed on the PV power plants and data collected from external sources, such as weather forecast services (NWP). While locally collected data are used to initialize part of the features in $w_{i,j,h}$ and the dependent variable $y_{i,j,h}$, external sources are used to initialize remaining features in $w_{i,j,h}$ (that are not collected by sensors), as well as features in $w'_{i,j,h}$. The raw data are preprocessed and normalized before subjecting them to the data mining algorithms. In this way, we solve problems related to measurement errors, null values and outliers, as explained in the following.

A. Locally Collected Data

The first problem we consider is how to deal with missing values related to sensor failures or communication problems for the features in $w_{i,j,h}$. It is noteworthy that most of the PV plants collect data every 10 or 15 minutes, thus we can still estimate reliable hourly averages if few measurement values are missing. However, if for a feature we cannot reliably compute its average due to the presence of several null values, we have to substitute them with a value returned by external systems for the specific position and time. To make the values collected by sensors comparable with the values returned by external systems for the same feature (e.g. temperature), they are both z-score normalized (see Equation (2)). If external systems provide us with no information, we replace the null value with the average of the feature observed for the same month of the same year at the same hour. Similarly, in the (rare) cases we are not able to reliably compute the hourly average for the independent variable $y_{i,j,h}$, we substitute it with the average value observed by the sensors in the same month at the same hour.

After replacing the missing values, we check for the presence of outliers: if the value of the feature x in $w_{i,j,h}$ observed by the sensors is outside the range $[\overline{x} - 4 \cdot \sigma_x; \overline{x} + 4 \cdot \sigma_x]$ (a relaxed 3-sigma rule), we consider it an outlier and handle it in the same way we handle null values.

B. External Data

In order to obtain external data, we query external services which provide interpolated data for past days and NWP values for the future days. Interpolated data are used either to initialize features in $w_{i,j,h}$ which are not collected by sensors, or, as previously mentioned, to correct incomplete and wrong data (in $w_{i,j,h}$) obtained by the sensors. On the contrary, NWP values are used to initialize features in $w'_{i,j,h}$.

Independently of the use of data obtained from external sources, the input parameters we use for querying data are latitude, longitude and time-stamp of interest. The obtained features are: pressure, percentage of cloud cover, type of precipitation, intensity of precipitation, temperature, dew point, ozone, wind speed, humidity, wind bearing and irradiance. Once retrieved, all the values are z-score normalized (see (2)).

Finally, in order to appropriately learn a prediction model (especially for ANNs), data must be scaled to the unit interval. Hence, we apply a min-max normalization [14] for each final feature, considering the *min* and *max* of the values observed (for $w_{i,j,h}$ and $w'_{i,j,h}$, obtained after z-score normalization). Actually, we consider the *max* increased by 30 percent, to handle future situations in which the observed values of each feature might exceed the current maximum.

V. EXPERIMENTS

A. Datasets

In our empirical evaluation, we consider two datasets: a real dataset, named PVItaly, collected by an Italian company, and a dataset concerning the PV production in USA available at the National Renewable Energy Innovation (NREL) web site (http://www.nrel.gov/), and henceforth referred to as NREL.

PVItaly data are collected at regular intervals of 15 minutes (measurements start at 2:00 and stop at 20:00 every day) by sensors located on 18 plants in Italy. The time period spans from January 1st, 2012 to May 4th, 2014. The installed peak power of the PV arrays is between 982.80 KW peak and 999.99 KW peak, and the average is 995.71 KW peak. The amount of missing values and outliers in the data is relatively small, if compared to the total amount of data: around 1% for missing values and around 3% for outliers.

The NREL dataset originally consists of simulated PV data for 6000 plants for the year 2006. We perform cluster sampling over the original dataset by first selecting 16 States with the

TABLE II
INDEPENDENT VARIABLES USED IN DIFFERENT SCENARIOS. h IS NOT USED IN THE DAILY SETTING.

	Non-temporal	Non-cyclic	Cyclic
NoSpat.	p, α, w	j, h, p, α, w	j, h, t, p, α, w
LatLon	p, α, w, π	j, h, p, α, w, π	j, h, t, p, α, w, π
LISA	p, α, w, s^{lisa}	j, h, p, α, w, s^{lisa}	j, h, t, p, α, w, s^{lisa}
PCNM	p, α, w, s^{pcnm}	j, h, p, α, w, s^{pcnm}	j, h, t, p, α, w, s^{pcnm}

highest Global Horizontal Irradiation (GHI). Then, from each State, we select 3 PV plants, resulting in PV data from 48 plants. The plant capacity is between 7 MW and 200 MW, and the average is 82.37 MW.

The weather data is queried from Forecast.io (http://forecast.io/), while the irradiance for the PVItaly dataset is queried from PVGIS (http://re.jrc.ec.europa.eu/pvgis/apps4/pvest.php). Forecast.io data comes from a wide range of data sources, which are statistically aggregated to provide the most accurate forecast possible for a given location. PVGIS data makes use of monthly averages of daily sums of global and diffuse irradiation. The averages represent the period 1981-1990. We queried the PVGIS database for each day and plant separately by using a custom-made wrapper, specifying the date and the coordinates of the plants of interest (latitude and longitude).

More formally, for the defined vectors in Section III, the following input features are considered:

- π_i: latitude, longitude;
- j, h: day and hour, respectively;
- $\alpha_{i,j,h}$: altitude and azimuth, queried from SunPosition (http://www.susdesign.com/sunposition/index.php);
- p_i: site ID, brand ID, model ID, age in months;
- $w_{i,j,h}$ and $w'_{i,j,h}$: ambient temperature, irradiance, pressure, wind speed, wind bearing, humidity, dew point, cloud cover, descriptive weather summary;
- $s^{lisa}_{i,j,h}$: LISA indexes $I^{(x)}_{i,j,h}$ for each variable x from $w_{i,j,h}$ (or $w_{i,j,h}$) and $\alpha_{i,j,h}$;
- s^{pcnm}_{i}: n PCNM coordinates (n = 15 for the PVItaly dataset, n = 30 for the NREL dataset),
- $t_{i,j,h}$: radial day distance d_r and radial representation of the hour h_r.

All datasets, the results and the system are available at: http://www.di.uniba.it/~ceci/energyprediction/.

B. Experimental Settings

We distinguish between hourly and daily settings. In the hourly setting, we investigate non-structured models with single output - the production $y_{i,j,h}$ of the plant P_i at a specified day j and specified hour h. In the daily setting, we investigate structured models with 24 outputs - the productions $y_{i,j}$ of the plant P_i for the hours from 1:00 to 24:00 on a specified day j (actually, we only consider the interval 2:00-20:00, because of the available data). For both settings, we investigate several scenarios with increasing spatio/temporal complexity (see Table II). Moreover, in the daily scenarios, we consider the representation summarized in Table I. The number of attributes and the number of examples considered for each scenario for the both datasets are given in Table III.

TABLE III
NUMBER OF ATTRIBUTES PER SCENARIO. N: NUMBER OF EXAMPLES.

PVItaly	Hourly (N = 276 811)			Daily (N = 14 569)		
	Non-temporal	Non-cyclic	Cyclic	Non-temporal	Non-cyclic	Cyclic
NoSpat.	15	18	19	230	233	234
LatLon	17	20	21	232	235	236
LISA	26	29	30	420	423	424
PCNM	30	33	34	245	248	249
NREL	Hourly (N = 331 968)			Daily (N = 17 520)		
	Non-temporal	Non-cyclic	Cyclic	Non-temporal	Non-cyclic	Cyclic
NoSpat.	12	14	15	209	211	212
LatLon	14	16	17	211	213	214
LISA	21	23	24	380	382	383
PCNM	42	44	45	239	241	242

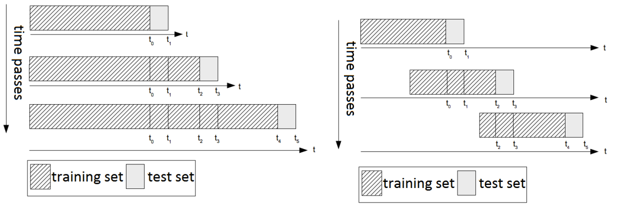


Fig. 4. Evaluation (training and testing) procedure with landmark window model (left) and count-based sliding window model (right).

For the evaluation, the datasets are randomly split into training days (85%) and testing days (15%). The learning strategy is iterative - for each testing day, the model is learned on all the previous days and tested on the considered day (example(s) unseen by the trained model). After testing, the testing day becomes part of the training set. This testing-retraining procedure is repeated for each testing day and the error contributes to the reported result. The evaluation procedure is based on the "landmark window model" and on the "count-based sliding window model" [11]. While the first model takes into account all the historical data, the second only takes into account the most recent window of a given number of instances. We performed experiments with the count-based sliding window model for window lengths 1000, 2000 and 4000 examples (see Fig. 4). We also investigate different values for the $maxDist$ threshold for the LISA method. Experiments are run three times with different random splits into training and testing sets, and the average error on the test set is reported.

For ANNs, we use the *encog* implementation of the Resilient Propagation (RPROP+) algorithm for training neural networks (http://www.heatonresearch.com/wiki/Resilient_Propagation#Implementing_RPROP.2B). RPROP+ is one of the best general-purpose neural network training methods implementing the back-propagation technique. It performs a direct adaptation of the weight step based on local gradient information. The basic principle 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. We use RPROP+ since it has been proven effective for renewable energy prediction [4]. Furthermore, RPROP+ does not require typical ANN parameters, such as learning rate and momentum, since they are automatically tuned during the training phase. The ANN topology has 1 hidden layer with the number of hidden neurons equal to 2/3 of the sum of the number of inputs and outputs.

For regression trees, we use the system CLUS that views a tree as a hierarchy of clusters (Predictive Clustering Trees - PCTs): the top-node corresponds to one cluster containing all the data, which is recursively partitioned into smaller clusters while moving down the tree. CLUS, including PCTs for multi-target regression [16], is available at clus.sourceforge.net.

For the evaluation of the results, we consider three indicators of the predictive performance, namely, the Root Mean Squared Error (RMSE), the Mean Absolute Error (MAE) (MAE results are available at www.di.uniba.it/~ceci/energyprediction/) and the improvement with respect to the persistence model (i.e., the model that forecasts the same production observed 24 hours before). For an analysis at a disaggregated level, we also perform feature selection with the aim of automatically identifying the most relevant input features for the task at hand. According to [13], the feature selection step has been performed as a best-first search in backward mode: starting from the complete feature set, the worth of each subset of attributes is evaluated by considering the individual predictive ability of each feature along with their degree of redundancy.

C. Results and Discussion

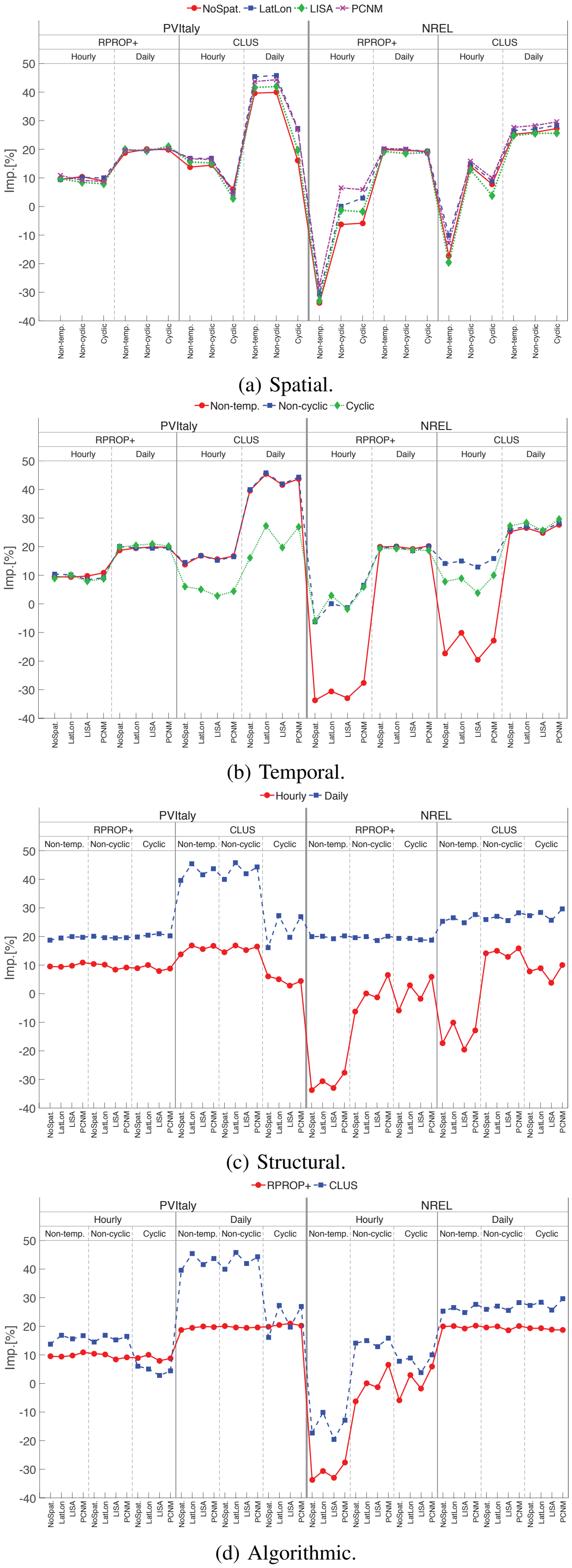
The results on the PVItaly and NREL datasets for the investigated hourly and daily scenarios are reported in Table IV for RPROP+ and CLUS. Negative percentages of improvement mean that the investigated model does not outperform the persistence model. The improvement of the best performing results are highlighted in bold. The results show that the best results for the PVItaly dataset are obtained in the setting LISA, Non-cyclic, CLUS, Daily; while for the NREL dataset the best results are obtained in the setting PCNM, Cyclic, CLUS, Daily. Moreover, from the results, it is clear that the landmark model outperforms the count-based sliding window model. This is confirmed also statistically (see Table V, sections "TRAINING WINDOW"). Finally, although there is no statistical evidence, LISA with small values of $maxDist$ show the best performances for PVItaly. This is not confirmed in NREL, where distances between plants are larger and behaviors at coarse-grained granularity are caught by the models.

In addition to the results reported in the tables, we also perform a more systematic analysis in order to understand what matters if we want to achieve good and reliable predictions. According to the contributions stated in the introduction, we compare the results across four dimensions:

1) *Spatial*: NoSpat. vs. LatLon vs. LISA vs. PCNM;
2) *Temporal*: Non-temporal vs. Non-cyclic vs. Cyclic;
3) *Structural*: Hourly vs. Daily;
4) *Algorithmic*: RPROP+ vs. CLUS.

All the above comparisons are orthogonally performed on both real-world datasets. The different dimensions of analysis for the landmark window model are graphically presented in Fig. 5. Additionally, the analysis is complemented with statistical (Wilcoxon signed rank) tests (see Table V). The considered LISA results are for $maxDist$ of 15 km for the PVItaly dataset and 600 km for the NREL dataset.

1) Spatial: The comparison in Fig. 5 (a) shows that, by varying the spatial complexity, there is no single spatial configuration which outperforms all the others. However, by considering the statistical tests for spatial autocorrelation, we can see that PCNM is the best performing method. The



(a) Spatial.

(b) Temporal.

(c) Structural.

(d) Algorithmic.

Fig. 5. Different dimensions of analysis for the landmark window model.

TABLE IV

RPROP+ and CLUS Average Performance (3 runs) on the PVItaly and NREL datasets. For LISA, a,b and c indicate different values of $maxDist$: 15 km, 30 km and 45 km for PVItaly, and 300 km, 600 km and 900 km for NREL. S1000, S2000 and S4000 indicate the sliding count-based model with different window lengths.

RPROP+	**PVItaly Dataset**												**NREL Dataset**											
	Hourly						**Daily**						**Hourly**						**Daily**					
	Non-temp.		**Non-cyclic**		**Cyclic**		**Non-temp.**		**Non-cyclic**		**Cyclic**		**Non-temp.**		**Non-cyclic**		**Cyclic**		**Non-temp.**		**Non-cyclic**		**Cyclic**	
S1000	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%
NoSpatial	0.139	-9.2	0.141	-10.4	0.140	-9.6	0.114	10.67	0.112	**12.07**	0.115	9.85	0.172	-42.4	0.139	-14.6	0.148	-21.9	0.123	-1.8	0.129	-6.6	0.130	-7.1
LatLon	0.137	-7.8	0.139	-9.5	0.141	-10.9	0.112	11.91	0.113	11.13	0.115	9.80	0.168	-38.3	0.131	-8.4	0.140	-15.8	0.127	-5.0	0.126	-3.7	0.147	-21.4
PCNM	0.147	-15.5	0.148	-16.3	0.149	-17.2	0.114	10.47	0.112	12.00	0.114	10.22	0.175	-44.6	0.139	-15.1	0.148	-22.6	0.112	7.54	0.112	7.59	0.117	3.64
LISA A	0.139	-9.5	0.140	-9.9	0.140	-10.3	0.124	2.76	0.124	2.61	0.128	-0.4	0.131	-8.0	0.121	0.39	0.121	0.27	0.105	13.60	0.101	16.47	0.107	11.89
LISA B	0.144	-12.9	0.143	-12.2	0.144	-13.3	0.126	0.65	0.126	1.12	0.131	-2.7	0.172	-42.3	0.135	-11.5	0.149	-23.1	0.116	4.51	0.123	-1.2	0.129	-6.5
LISA C	0.144	-12.7	0.143	-12.7	0.124	2.55	0.125	1.57	0.143	-12.6	0.126	0.74	0.130	-7.1	0.121	-0.2	0.118	2.32	0.102	15.55	0.105	13.17	0.101	**17.02**
S2000	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%
NoSpatial	0.136	-7.0	0.137	-7.7	0.138	-8.7	0.114	10.22	0.113	11.54	0.115	9.77	0.166	-37.3	0.132	-8.9	0.127	-4.9	0.130	-7.2	0.128	-5.9	0.128	-6.0
LatLon	0.137	-7.3	0.140	-9.6	0.139	-9.0	0.115	9.49	0.112	**12.22**	0.114	10.20	0.163	-34.6	0.123	-1.4	0.123	-1.5	0.128	-5.8	0.131	-7.9	0.124	-2.2
PCNM	0.144	-13.4	0.146	-15.0	0.149	-17.1	0.114	10.52	0.115	9.58	0.116	9.17	0.163	-34.4	0.128	-6.0	0.127	-5.1	0.120	1.11	0.120	0.63	0.117	3.36
LISA A	0.135	-6.4	0.137	-8.0	0.138	-8.2	0.124	2.84	0.123	3.35	0.126	0.99	0.120	1.16	0.116	3.84	0.120	1.30	0.114	5.59	0.113	6.94	0.128	-5.3
LISA B	0.136	-7.1	0.139	-9.5	0.140	-9.8	0.128	-0.2	0.128	-0.7	0.128	-0.6	0.166	-36.8	0.130	-7.3	0.129	-6.4	0.134	-10.7	0.126	-4.0	0.132	-9.2
LISA C	0.139	-8.9	0.137	-7.6	0.141	-11.0	0.126	1.35	0.125	1.83	0.126	1.09	0.125	-3.1	0.114	5.78	0.116	4.24	0.111	**8.27**	0.112	7.67	0.119	1.77
S4000	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%
NoSpatial	0.130	-1.7	0.133	-4.7	0.134	-5.0	0.108	**14.91**	0.109	14.03	0.109	14.35	0.162	-33.8	0.125	-3.5	0.127	-5.1	0.124	-2.6	0.126	-4.0	0.126	-3.9
LatLon	0.135	-6.1	0.133	-4.3	0.135	-6.2	0.110	13.69	0.110	13.43	0.112	12.11	0.157	-29.5	0.119	1.48	0.123	-1.5	0.125	-3.4	0.124	-2.6	0.137	-13.1
PCNM	0.145	-13.7	0.145	-14.0	0.142	-11.4	0.110	13.88	0.112	12.40	0.111	12.67	0.157	-29.2	0.120	1.19	0.129	-6.3	0.115	5.10	0.117	3.08	0.123	-1.3
LISA A	0.134	-5.3	0.135	-5.8	0.132	-4.0	0.116	8.85	0.118	7.03	0.117	8.30	0.112	7.79	0.105	**13.18**	0.107	11.35	0.121	-0.2	0.112	7.30	0.115	5.45
LISA B	0.135	-5.7	0.137	-7.8	0.138	-8.2	0.124	2.28	0.123	3.00	0.121	5.23	0.158	-30.7	0.123	-1.8	0.130	-7.3	0.129	-6.4	0.132	-8.6	0.135	-11.1
LISA C	0.138	-8.6	0.137	-7.9	0.138	-8.6	0.115	9.56	0.123	3.78	0.117	8.33	0.115	4.74	0.114	5.85	0.109	9.67	0.115	4.89	0.115	4.98	0.109	10.07
Landm.	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%
NoSpatial	0.115	9.50	0.114	10.44	0.116	8.94	0.103	18.73	0.102	20.02	0.102	19.85	0.162	-33.5	0.129	-6.3	0.128	-5.8	0.097	19.65	0.097	19.58	0.098	19.29
LatLon	0.115	9.37	0.114	10.15	0.115	10.00	0.102	19.51	0.102	19.63	0.101	20.49	0.159	-31.0	0.121	0.10	0.118	2.98	0.097	20.07	0.097	19.91	0.098	19.32
PCNM	0.113	10.86	0.116	9.13	0.116	8.77	0.102	19.77	0.102	19.63	0.102	20.21	0.155	-27.7	0.113	6.55	0.112	7.34	0.097	**20.24**	0.097	20.14	0.098	19.49
LISA A	0.115	9.71	0.117	8.44	0.117	7.93	0.102	19.91	0.103	19.44	0.101	**20.99**	0.115	5.25	0.105	13.24	0.106	12.78	0.099	18.39	0.107	11.44	0.11	10.14
LISA B	0.125	1.67	0.128	-0.1	0.126	1.27	0.113	11.19	0.117	8.26	0.117	8.28	0.161	-32.9	0.123	-1.3	0.122	-1.1	0.098	19.26	0.099	18.54	0.097	19.95
LISA C	0.125	1.91	0.128	-0.7	0.126	0.70	0.115	9.89	0.113	10.86	0.118	7.71	0.113	6.34	0.110	9.55	0.118	2.65	0.122	-0.3	0.103	15.15	0.11	9.44
CLUS	**PVItaly Dataset**												**NREL Dataset**											
	Hourly						**Daily**						**Hourly**						**Daily**					
	Non-temp.		**Non-cyclic**		**Cyclic**		**Non-temp.**		**Non-cyclic**		**Cyclic**		**Non-temp.**		**Non-cyclic**		**Cyclic**		**Non-temp.**		**Non-cyclic**		**Cyclic**	
S1000	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%
NoSpatial	0.111	12.85	0.110	13.93	0.133	-4.6	0.074	42.17	0.073	42.32	0.103	19.49	0.157	-29.9	0.114	5.76	0.137	-13.1	0.098	19.36	0.090	25.91	0.089	26.36
LatLon	0.108	15.45	0.107	16.11	0.131	-3.0	0.066	47.85	0.066	47.99	0.099	22.50	0.145	-19.7	0.113	6.76	0.133	-10.2	0.096	20.68	0.088	27.02	0.089	26.66
PCNM	0.108	15.27	0.108	15.36	0.130	-2.1	0.068	46.61	0.068	46.86	0.098	23.17	0.149	-23.0	0.112	7.63	0.136	-12.4	0.089	26.82	0.087	28.29	0.086	**28.92**
LISA A	0.109	14.59	0.109	14.77	0.133	-4.3	0.071	44.17	0.071	44.52	0.102	19.84	0.174	-43.3	0.139	-14.4	0.168	-39.0	0.109	9.92	0.111	8.15	0.112	7.41
LISA B	0.105	17.54	0.104	18.00	0.130	-2.0	0.063	50.36	0.063	**50.62**	0.098	22.84	0.160	-32.3	0.116	4.34	0.144	-19.0	0.093	23.26	0.090	25.51	0.090	25.98
LISA C	0.105	17.73	0.104	17.98	0.132	-3.9	0.064	49.94	0.063	50.22	0.101	20.76	0.175	-44.7	0.138	-13.9	0.157	-29.3	0.114	6.08	0.111	8.00	0.111	8.08
S2000	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%
NoSpatial	0.111	12.88	0.110	13.38	0.132	-3.5	0.075	41.22	0.074	41.58	0.103	18.86	0.157	-29.9	0.114	5.76	0.129	-6.7	0.098	19.36	0.090	25.91	0.088	27.11
LatLon	0.107	15.66	0.107	15.58	0.129	-1.4	0.068	46.66	0.067	47.21	0.098	22.98	0.145	-19.7	0.113	6.76	0.127	-4.5	0.096	20.68	0.088	27.02	0.087	28.08
PCNM	0.108	15.29	0.108	15.14	0.129	-1.2	0.070	45.32	0.069	46.00	0.098	23.27	0.149	-23.0	0.112	7.63	0.127	-5.2	0.089	26.82	0.087	28.29	0.085	**29.90**
LISA A	0.108	15.17	0.108	15.12	0.131	-2.6	0.072	43.42	0.072	43.72	0.103	19.41	0.182	-49.9	0.139	-14.4	0.158	-30.6	0.112	7.36	0.111	8.15	0.104	14.18
LISA B	0.105	17.60	0.105	17.88	0.134	-5.1	0.064	49.76	0.063	**50.19**	0.109	14.50	0.160	-32.3	0.116	4.34	0.116	4.34	0.093	23.26	0.090	25.51	0.090	25.51
LISA C	0.105	17.26	0.105	17.31	0.130	-1.8	0.064	49.61	0.064	49.58	0.101	20.84	0.175	-44.7	0.138	-13.9	0.157	-29.3	0.114	5.88	0.111	8.01	0.111	8.08
S4000	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%
NoSpatial	0.111	12.70	0.110	13.57	0.130	-2.4	0.076	40.68	0.075	40.92	0.101	20.86	0.157	-29.9	0.114	5.76	0.122	-0.6	0.098	19.36	0.090	25.91	0.088	27.29
LatLon	0.108	15.36	0.107	15.70	0.128	-0.9	0.068	46.37	0.068	46.60	0.095	25.09	0.145	-19.7	0.113	6.76	0.121	0.34	0.096	20.68	0.088	27.02	0.087	28.25
PCNM	0.108	15.25	0.108	15.20	0.129	-1.2	0.070	45.09	0.069	45.45	0.095	25.28	0.149	-23.0	0.112	7.63	0.119	1.46	0.089	26.82	0.087	28.29	0.085	**29.73**
LISA A	0.109	14.43	0.109	14.39	0.129	-1.4	0.073	42.96	0.072	43.12	0.101	20.59	0.182	-49.9	0.139	-14.4	0.149	-23.1	0.112	7.36	0.111	8.15	0.111	8.39
LISA B	0.112	11.98	0.112	12.27	0.136	-7.2	0.074	41.56	0.075	41.05	0.105	17.87	0.160	-32.3	0.116	4.34	0.127	-4.6	0.093	23.26	0.090	25.51	0.088	27.06
LISA C	0.105	17.53	0.104	17.97	0.138	-8.6	0.068	**46.70**	0.126	1.22	0.100	21.36	0.175	-44.7	0.138	-13.9	0.157	-29.3	0.114	5.88	0.111	8.00	0.111	8.08
Landm.	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%	RMSE	Imp.%
NoSpatial	0.110	13.78	0.109	14.43	0.120	6.05	0.077	39.64	0.077	39.90	0.107	16.04	0.142	-17.3	0.104	14.07	0.112	7.81	0.091	25.24	0.090	25.96	0.088	27.27
LatLon	0.106	16.82	0.106	16.85	0.121	5.10	0.070	45.37	0.069	45.80	0.093	27.27	0.133	-10.1	0.103	14.98	0.110	8.95	0.089	26.52	0.088	27.07	0.087	28.44
PCNM	0.106	16.69	0.107	16.44	0.122	4.38	0.072	43.70	0.071	44.32	0.093	26.97	0.137	-12.8	0.102	15.85	0.109	9.97	0.088	27.72	0.087	28.33	0.085	**29.60**
LISA A	0.108	15.46	0.108	15.11	0.121	4.83	0.074	41.57	0.074	41.86	0.105	17.68	0.182	-49.9	0.138	-13.9	0.139	-14.7	0.112	7.36	0.111	7.94	0.111	8.04
LISA B	0.115	10.04	0.114	10.77	0.134	-5.2	0.074	41.77	0.074	41.56	0.102	19.95	0.145	-19.6	0.106	12.82	0.113	6.77	0.091	24.71	0.090	25.51	0.088	27.10
LISA C	0.108	15.52	0.107	15.73	0.122	4.16	0.067	47.00	0.067	**47.24**	0.095	25.36	0.175	-44.7	0.138	-13.9	0.134	-10.8	0.114	5.88	0.111	8.00	0.110	9.06

PCNM method outperforms all other spatial configurations (significantly outperforming NoSpat. and LISA). The LISA method is not able to even outperform the simple LatLon configuration. This means that modeling autocorrelation by considering the spatial structure of the data (as PCNM does) is, in the application at hand, much more important than directly considering auto-regressive information (as LISA does). Finally, what is clear from the results is that considering spatial autocorrelation (in any form) is generally beneficial (LatLon, LISA and PCNM outperform NoSpat.).

2) Temporal: By analyzing the contribution of temporal autocorrelation, we can see that results are not uniform in the sense that there is no single temporal configuration which outperforms the others over the two learning settings, the two learning algorithms, and the two datasets. However, the statistical tests show that the two configurations which exploit

TABLE V
p-VALUES OF THE SIGNED WILCOXON RANK TESTS FOR DIFFERENT DIMENSIONS OF ANALYSIS. IN BOLD STATISTICALLY SIGNIFICANT VALUES (CONFIDENCE=0.05, UNLESS SPECIFIED OTHERWISE).

		p-value	winner
SPATIAL (with Bonferroni correction $< 0.05/6$)	NoSpat. VS LatLon	**0.0002**	LatLon
	NoSpat. VS LISA	0.954	LISA
	NoSpat. VS PCNM	**0.0008**	PCNM
	LatLon VS LISA	**3.43E-05**	LatLon
	LatLon VS PCNM	0.775	PCNM
	LISA VS PCNM	**4.67E-05**	PCNM
TEMPORAL (with Bonferroni correction $< 0.05/3$)	Non-temp. VS Non-cyclic	**0.005**	Non-cyclic
	Non-temp. VS Cyclic	0.594	Cyclic
	Non-cyclic VS Cyclic	**0.007**	Non-cyclic
STRUCTURAL	Hourly VS Daily	**5.22E-09**	Daily
ALGORITHMIC	RPROP+ VS CLUS	**4.39E-08**	CLUS
TRAINING WINDOW (RPROP+) (with B. corr. $< 0.05/3$	Sliding 1000 vs Landmark	**8.15E-10**	Landmark
	Sliding 2000 vs Landmark	**8.68E-10**	Landmark
	Sliding 4000 vs Landmark	**4.63E-09**	Landmark
TRAINING WINDOW (CLUS) (with B. corr. $< 0.05/3$)	Sliding 1000 vs Landmark	**1.96E-04**	Landmark
	Sliding 2000 vs Landmark	**2.44E-04**	Landmark
	Sliding 4000 vs Landmark	**9.83E-05**	Landmark

temporal autocorrelation (Non-cyclic and Cyclic) significantly outperform the configuration that does not (Non-temporal), with the cyclic configuration not outperforming the non-cyclic one. This can be due to collinearity problems because of high correlation between cyclic temporal features and non-cyclic ones. Moreover, since in two out of four daily settings, the best improvement is obtained with cyclic configuration, we conclude that temporal autocorrelation is properly exploited in the case of structured output prediction, where dependencies between the productions at two different hours of the same day provide a sort of "contiguity" with respect to the temporal autocorrelation captured by the directional statistics.

3) Structural: By comparing the non-structured output prediction setting with structured output prediction, we can clearly see that the comparison is significantly in favor of the structured output prediction setting, confirming that dependence between the predictions obtained at different hours of the same day is of fundamental importance for improving predictions.

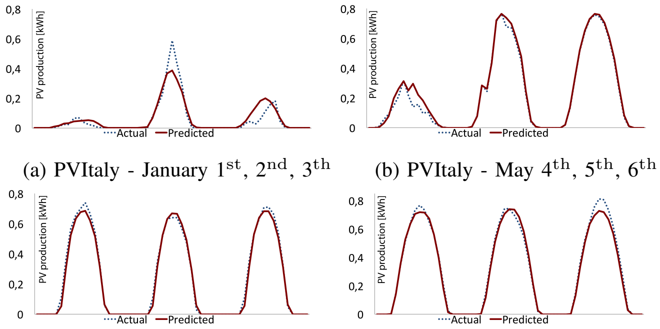
4) Algorithmic: CLUS outperforms RPROP+ with a great margin, especially for the structured output prediction setting, where it is able to improve the predictions of the persistent model for more than 45% in the case of PVItaly (with the combination Non-cyclic - LatLon) and of almost 30% in the case of NREL (with the combination Cyclic - PCNM).

Concerning feature selection, the results reported in Table VI show that it does not lead to improved performances. This means that many and distinct features contribute to provide information for better predictions. However, from the application of the feature selection algorithm, we can still have additional insights. In fact, the selected features confirm that the added spatial and temporal features are considered important for the prediction. For PVItaly, the selected features are: day, date, hour (hourly setting), irradiance, humidity, irradianceLISA, azimuthLISA, while for NREL they are: date, day, hour (hourly setting), longitude, windspeed, altitude, azimuth, cloudcoverLISA, azimuthLISA, PCNM27, PCNM28.

A different, better marked perspective of the results is provided in Fig.6 where we compare the actual and predicted curves of the production for 3 consecutive winter and summer days for both datasets for the landmark window model. For the PVItaly dataset example, a decrease in predictive performances can be observed under cloudy and rainy weather conditions (typical for winter days) compared to sunny weather conditions (typical for summer days). Moreover, the PVItaly dataset has shown to be more challenging than NREL. Namely, the PVItaly is real-world dataset, while the NREL dataset is simulated. Therefore, it is not surprising for PVItaly to exhibit greater oscillations than NREL.

TABLE VI
AVERAGE RMSE (OVER 3 RUNS) WITH AND WITHOUT FEATURE SELECTION (FS). RESULTS WITHOUT FEATURE SELECTION ARE OBTAINED WITH THE CONFIGURATION PCNM - NON-CYCLIC.

	RPROP		CLUS	
PVItaly	With FS	Without FS	With FS	Without FS
Hourly	0.121	0.116	0.113	0.107
Daily	0.117	0.102	0.083	0.071
NREL	With FS	Without FS	With FS	Without FS
Hourly	0.127	0.113	0.115	0.102
Daily	0.122	0.097	0.105	0.087



(a) PVItaly - January 1^{st}, 2^{nd}, 3^{th} (b) PVItaly - May 4^{th}, 5^{th}, 6^{th}

(c) NREL - January 16^{th}, 17^{th}, 18^{th} (d) NREL - May 2^{nd}, 3^{th}, 4^{th}

Fig. 6. Predicted (solid line) vs. actual (dotted line) production for three consecutive days (in January and May) of a single plant of PVItaly and NREL. Results for PVItaly are obtained with the CLUS-Daily-LatLon-Non-cyclic configuration, while for NREL are obtained with the CLUS-Daily-PCNM-Non-cyclic configuration. The time intervals considered are 2:00 AM - 8:00 PM. The considered plant for PVItaly is located in Sannicandro di Bari, Italy at latitude: 40.984261, longitude: 16.831031; the considered plant for NREL is located in Riverside, California at latitude: 33.671418, longitude: -115.558126.

VI. CONCLUSIONS

This paper studies the problem of PV energy prediction by considering different dimensions of analysis: spatio/temporal autocorrelation, the learning setting (structured output vs. non-structured output) and the learning algorithm (ANNs vs. regression trees), aiming to investigate the relevant aspects for the problem at hand. Results clearly show that structured output prediction models are much more accurate than models that predict single outputs because structured output prediction models can capture dependencies between different hours of the same day. Moreover, experimental results confirm that both forms of autocorrelation should be taken into account in this specific application: While PCNM is the best way to consider spatial autocorrelation, the simple consideration of hour and day information is enough to properly catch temporal autocorrelation. Finally, regression trees produce significantly better predictions than ANNs, indicating that also in PV energy prediction, hierarchical models are better than flat regression functions (this was also observed in [24] for predicting electricity energy consumption).

As future work, we intend to directly consider autocorrelation in the cost function (heuristic) used to learn the ANN. We also plan to investigate more sophisticated learning methods based on ensemble techniques.

References

[1] L. Anselin, “Local indicators of spatial association lisa,” Geographical Analysis, vol. 27, no. 2, pp. 93–115, 1995.

[2] ——, “Spatial dependence in linear regression models with an introduction to spatial econometrics,” 1996, literaturverz. S. 43 - 53.

[3] P. Bacher, H. Madsen, and H. A. Nielsen, “Online short-term solar power forecasting,” Solar Energy, vol. 83, no. 10, pp. 1772 – 1783, 2009.

[4] R. Bessa, V. Miranda, and J. Gama, “Entropy and correntropy against minimum square error in offline and online three-day ahead wind power forecasting,” Power Systems, IEEE Transactions on, vol. 24, no. 4, pp. 1657–1666, 2009.

[5] S. Bofinger and G. Heilscher, “Solar electricity forecast - approaches and first results,” in 20th Europ. PV conf., 2006.

[6] D. Borcard, P. Legendre, C. Avois-Jacquet, and H. Tuomisto, “Dissecting the spatial structure of ecological data at multiple scales,” Ecology, vol. 85, no. 7, pp. 1826–1832, Jul. 2004.

[7] S. Buhan and I. Cadirci, “Multistage wind-electric power forecast by using a combination of advanced statistical methods,” IEEE Trans. Industrial Informatics, vol. 11, no. 5, pp. 1231–1242, 2015.

[8] P. Chakraborty, M. Marwah, M. F. Arlitt, and N. Ramakrishnan, “Fine-grained photovoltaic output prediction using a bayesian ensemble,” in AAAI, 2012.

[9] S. Dray, P. Legendre, and P. R. Peres-Neto, “Spatial modelling: a comprehensive framework for principal coordinate analysis of neighbour matrices (pcnm),” Ecological Modelling, vol. 196, no. 34, pp. 483 – 493, 2006.

[10] M. M. Gaber, A. Zaslavsky, and S. Krishnaswamy, “Mining data streams: A review,” SIGMOD Rec., vol. 34, no. 2, pp. 18–26, Jun. 2005.

[11] J. Gama and M. M. Gaber, Eds., Learning from Data Streams. Springer, 2007.

[12] P. M. Gonçalves Jr and R. S. De Barros, “Rcd: A recurring concept drift framework,” Pattern Recogn. Lett., vol. 34, no. 9, pp. 1018–1025, 2013.

[13] M. A. Hall, “Correlation-based feature subset selection for machine learning,” Ph.D. dissertation, University of Waikato, Hamilton, New Zealand, 1998.

[14] J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, 3rd ed. Morgan Kaufmann, 2011, pp. 113–114.

[15] J. Kleissl, Solar Resource Assessment and Forecasting. Elsevier, 2013.

[16] D. Kocev, C. Vens, J. Struyf, and S. Džeroski, “Tree ensembles for predicting structured outputs,” Pattern Recognition, vol. 46, no. 3, pp. 817–833, 2013.

[17] I. Kühn, “Incorporating spatial autocorrelation may invert observed patterns,” Diversity and Distributions, vol. 13, no. 1, pp. 66–69, 2007.

[18] P. Mathiesen and J. Kleissl, “Evaluation of numerical weather prediction for intra-day solar forecasting in the continental united states,” Solar Energy, vol. 85, no. 5, pp. 967–977, 2011.

[19] S. Pelland, G. Galanis, and G. Kallos, “Solar and photovoltaic forecasting through post-processing of the global environmental multiscale numerical weather prediction model,” Prog Photovolt Res Appl, vol. 21, no. 3, pp. 284–296, 2013.

[20] A. Rashkovska, J. Novljan, M. Smolnikar, M. Mohorčič, and C. Fortuna, “Online short-term forecasting of photovoltaic energy production,” in Innovative Smart Grid Technologies Conference (ISGT), 2015 IEEE Power & Energy Society, 2015, pp. 1–5.

[21] N. Sharma, P. Sharma, D. E. Irwin, and P. J. Shenoy, “Predicting solar generation from weather forecasts using machine learning,” in SmartGridComm. IEEE, 2011, pp. 528–533.

[22] D. Stojanova, M. Ceci, A. Appice, and S. Dzeroski, “Network regression with predictive clustering trees,” Data Min. Knowl. Discov., vol. 25, no. 2, pp. 378–413, 2012.

[23] M. Szummer and R. W. Picard, “Temporal texture modeling,” in IEEE Intl. Conf. Image Processing, vol. 3, Sep. 1996, pp. 823–826.

[24] G. K. Tso and K. K. Yau, “Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks,” Energy, vol. 32, no. 9, pp. 1761 – 1768, 2007.

[25] H. H. Wagner and M. J. Fortin, “Spatial Analysis of Landscapes: Concepts and Statistics,” Ecology, vol. 86, no. 8, pp. 1975–1987, 2005.

Michelangelo Ceci, Ph.D., is an associate professor at the Department of Computer Science, University of Bari, Italy. His research interests are in data mining and machine learning. He has published more than 150 papers in reviewed journals and conferences. He is the unit coordinator of EU and national projects. He is in the Program Committee of many conferences, including: IEEE ICDM, IJCAI, ECML-PKDD, SIAM SDM, DS and ISMIS. He is in the editorial board of IJDSN, IJSNM, “Intelligenza Artificiale”, IJDATS and of the ECMLPKDD 2014-2016 journal tracks. He was program (co-)Chair of SEBD2007, Discovery Science 2016, and General Chair of ECML-PKDD2017.

Roberto Corizzo is a Ph.D. student at the Department of Computer Science, University of Bari, Italy. His research interests include Big Data analytics, data mining and predictive modeling techniques for sensor networks. He graduated at the University of Bari (MSc) with a thesis on multi-type clustering in heterogeneous networks. He has been involved in the development of algorithms for renewable energy power forecasting in the Vi-POC project.

Fabio Fumarola, Ph.D., is a research assistant at the Department of Computer Science, University of Bari, Italy. He was a visiting researcher at the University of Illinois and he founded a Startup working on Data Mining on Big Data. He published more than 20 papers in refereed journals and conferences. He has served in the Program Committee of several conferences, including: ECML-PKDD, Biocomputation, DS and ISMIS. His research interests include Big Data, Web Mining and Data Stream Mining.

Donato Malerba is a full professor at the Department of Computer Science, University of Bari, Italy. His research activity mainly concerns machine learning, data mining and Big Data. He published more than 200 papers in international journals and conference proceedings. He has been responsible for the local research unit of several European and National projects, and received an IBM Faculty Award in 2004. He is the Director of the Computer Science Department of the University of Bari and of the CINI Lab on Big Data. He is in the Board of Directors of the Big Data Value Association and in the Partnership Board of the PPP Big Data Value. He was Program (co-)Chair of IEA-AIE 2005, ISMIS 2006, SEBD 2007, ECMLPKDD 2011, and General Chair of ALT/DS 2016. He is in the editorial board of several international journals.

Aleksandra Rashkovska received her Ph.D. in Computer Science from the Jožef Stefan International Postgraduate School, Ljubljana, Slovenia in 2013. She is a research associate at the Department of Communication Systems, Jožef Stefan Institute, Ljubljana, Slovenia. She was a visiting researcher at the Department of Computer Science, University of Bari, Italy. Her research interests include advanced bio-signal analysis, computer simulations and data mining in biomedicine, and data mining in sensor networks.