ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8.
Available from http://www.i6doc.com/en/.

Solar PV Power Forecasting Using Extreme Learning Machine and Information Fusion

Hélène Le Cadre[1], Ignacio Aravena[2] and Anthony Papavasiliou[2]

1- MINES ParisTech, PSL Research university - CMA
CS 10207 rue Claude Daunesse, 06904 Sophia Antipolis Cedex - France

2- Université catholique de Louvain - CORE
Voie du Roman Pays 34, 1348 Louvain-la-Neuve - Belgium

Abstract. We provide a learning algorithm combining distributed Extreme Learning Machine and an information fusion rule based on the aggregation of experts advice, to build day ahead probabilistic solar PV power production forecasts. These forecasts use, apart from the current day solar PV power production, local meteorological inputs, the most valuable of which is shown to be precipitation. Experiments are then run in one French region, Provence-Alpes-Côte d'Azur, to evaluate the algorithm performance.

1 Motivation

Renewable energy integration in electric power systems is progressing at an unprecedented pace, particularly in Europe. The integration of renewable resources, however, poses several challenges to the current paradigm for operating power systems. One major challenge is to mitigate the effects of the unpredictability and volatility of renewable supply, which stems from its strong dependence on weather conditions.

Reserves i.e., capacity with the ability to be activated/deactivated in a short time interval, are used to handle uncertainty in power systems operations. They can be procured by flexible generation units and flexible loads in the system. Generally speaking, renewable energy integration drives system operators to increase the reserve margin. The determination of the reserve margin then becomes critical. An overestimation of reserves would result in excess online generation, which is expensive and also undermines the environmental benefits of renewable energy. An underestimation of reserves, on the other hand, can compromise system security. Currently, the process of determining reserves requirements is carried out based on ad-hoc rules, or long-term studies, in most systems.

Stochastic programming is a framework that relies on an explicit modeling of uncertainty in optimization problems. When applied to the short-term scheduling of power systems, it endogenously optimizes the required reserves for the time horizon to which it is applied, outperforming ad-hoc rules in the presence of both renewable energy and component failures [5]. The drawback of stochastic programming is that it requires a probabilistic description of the uncertain parameters, which is usually unavailable and difficult to estimate.

This paper presents an algorithm based on Extreme Learning Machine (ELM), that uses daily production data and local meteorological inputs to produce a 24

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8.
Available from http://www.i6doc.com/en/.

hour horizon probabilistic forecast of solar photovoltaic (PV) power production, at a regional scale. This forecast will be used later as an input for a study of the stochastic short-term scheduling of the European power system. The rest of the paper is organized as follows: in Section 2 we first give an overview of ELM. Then, we introduce probabilistic forecasting based on Prediction Interval (PI). Finally, we detail our algorithm combining ELM and experts advice fusion. The algorithm performance is evaluated in Section 3.

2 Proposed algorithm

Overview of Extreme Learning Machine (ELM): ELM is a single hidden layer feed-forward Neural Network which randomly selects the input weights and thresholds and determines the output weights through matrix computation. It is theoretically capable of approximating any continuous function and implementing any classification application [1], [7]. It avoids the traditional gradient based algorithms which are used in Back-Propagation Neural Network algorithms to tune the weights.

We focus on one French region: Provence-Alpes-Côte d'Azur (PACA), which is known for its high sunshine level. We let $(\mathbf{x_i})_{i=1,...,\text{size}_{Train}}$, with $\mathbf{x_i} \in \mathbb{R}^d, \forall i = 1, ..., \text{size}_{Train}$, be the training data[1]. In our experiments, we forecast the solar production on a 24 hour horizon basis with one prediction every 30 minutes taking as input the current day solar PV production with a granularity of 30 minutes and meteorological inputs measured locally by stations associated with the region. The output function of ELM is:

$$f_L(\mathbf{x}) = \sum_{j=1}^{L} \beta_{\mathbf{j}} h_j(\mathbf{x}) = h(\mathbf{x})\beta, \forall \mathbf{x} \in \mathbb{R}^d \tag{1}$$

where β is the (column) vector of the output weights between the hidden layer of L nodes to the output node and $h(\mathbf{x}) = \Big(h_1(\mathbf{x}), ..., h_L(\mathbf{x})\Big)$ is the output vector of the hidden layer with respect to the input $\mathbf{x}$. In our experiments detailed in Section 3, $h_j(\mathbf{x}) = g(\mathbf{w_j}.\mathbf{x} + b_j), \forall j = 1, ..., L$, will be associated with a sigmoid function where $\mathbf{w_j}$ is a weight vector and b_j is the threshold of the j-th hidden neuron. We let $\mathbf{H}$ be the considered region hidden layer output matrix: $\mathbf{H} = \begin{pmatrix} h(\mathbf{x_1}) \\ \vdots \\ h(\mathbf{x}_{\text{size}_{\mathbf{Train}}}) \end{pmatrix} = \begin{pmatrix} h_1(\mathbf{x_1}) & \dots & h_L(\mathbf{x_1}) \\ \vdots & \vdots & \vdots \\ h_1(\mathbf{x}_{\text{size}_{\mathbf{Train}}}) & \dots & h_L(\mathbf{x}_{\text{size}_{\mathbf{Train}}}) \end{pmatrix}$ and $\mathbf{T}$ be the region training data target matrix: $\mathbf{T} = \begin{pmatrix} t_1 \\ \vdots \\ t_{\text{size}_{Train}} \end{pmatrix}$. The input weights and thresholds being randomly chosen, training our ELM is equivalent

[1] We use the following classical notations: vectors/matrices will be written in bold letters and $\mathbf{x}.\mathbf{y}$ represents the scalar product of vectors $\mathbf{x}, \mathbf{y}$.

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8. Available from http://www.i6doc.com/en/.

to computing a solution of the following linear system : $\hat{\beta} = \mathbf{H}^{+}\mathbf{T}$ where $\mathbf{H}^{+}$ is the Penrose-Moore generalized inverse of matrix $\mathbf{H}$ [1], computed using Singular Value Decomposition (SVD).

Prediction Interval based ELM: We assume that the training data are corrupted by noise [7] which implies that the ELM output described by Equation (1) becomes: $\mathbf{t_i} = f_L(\mathbf{x_i}) + \epsilon(\mathbf{x_i})$ with $\epsilon(\mathbf{x_i}) \sim \mathcal{N}\Big(0; \sigma^2_\epsilon(\mathbf{x_i})\Big)$. ELM generates the averaged values of targets conditioned on input variable $\mathbf{x_i}$: $\hat{y}(\mathbf{x_i}) = f_L\Big(\mathbf{x_i}, (\hat{\mathbf{w}}_\mathbf{j})_{j=1,...,L}, (\hat{b_j})_{j=1,...,L}, \hat{\beta}\Big)$. Assuming that the error resulting from the estimation of the true regression and the other resulting from the estimation of the measured target are independent, we infer: $\sigma^2_t(\mathbf{x_i}) = \sigma^2_{\hat{y}}(\mathbf{x_i}) + \sigma^2_\epsilon(\mathbf{x_i})$. A $100(1-\alpha)\%$, with $\alpha \in [0;1]$, confidence level Prediction Interval (PI) of the measured target t_i is a stochastic interval defined by its lower and upper bounds: $L_\alpha(\mathbf{x_i}) = \hat{y}(\mathbf{x_i}) - z_{1-\frac{\alpha}{2}}\sqrt{\sigma^2_t(\mathbf{x_i})}$ and $U_\alpha(\mathbf{x_i}) = \hat{y}(\mathbf{x_i}) + z_{1-\frac{\alpha}{2}}\sqrt{\sigma^2_t(\mathbf{x_i})}$ where $z_{1-\frac{\alpha}{2}}$ is the corresponding critical value of the Gaussian density function. The PI construction is based on pairs Bootstrap [7], replicated B_M times, to obtain $\hat{y}(\mathbf{x_i})$ and $\sigma^2_{\hat{y}}(\mathbf{x_i})$. This leads us to: $\hat{y}(\mathbf{x_i}) = \frac{1}{B_M}\sum_{k=1}^{B_M}\hat{y}_k(\mathbf{x_i})$ and $\sigma^2_{\hat{y}}(\mathbf{x_i}) = \frac{1}{B_M-1}\sum_{k=1}^{B_M}\Big(\hat{y}_k(\mathbf{x_i}) - \hat{y}(\mathbf{x_i})\Big)^2$. To obtain the data noise variance, the Bootstrap is replicated B_N times using $h_\epsilon(\mathbf{x_i}) = \Big(\hat{y}(\mathbf{x_i}) - \mathbf{t_i}\Big)^2, \forall i = 1, ..., \text{size}_{Train}$, as output function. The output of the trained ELM being denoted $\hat{r}_k(\mathbf{x_i})$, we obtain: $\sigma^2_\epsilon(\mathbf{x_i}) = \frac{1}{B_N}\sum_{k=1}^{B_N}\hat{r}_k(\mathbf{x_i}) + \frac{1}{B_N-1}\sum_{k=1}^{B_N}\Big(\hat{r}_k(\mathbf{x_i}) - \frac{1}{B_N}\sum_{l=1}^{B_N}\hat{r}_l(\mathbf{x_i})\Big)$. We use two performance measures. PI coverage probability measures the forecaster reliability over each region: PICP $\triangleq \frac{1}{size_{Test}}\sum_{i=1}^{size_{Test}}\delta_i$ where $\delta_i = 1$ if $t_i \in [L_\alpha(\mathbf{x_i}); U_\alpha(\mathbf{x_i})]$; 0 otherwise. The Score, measured as the PI interval width, captures the forecaster sharpness over the region: Score $\triangleq \Big(U_\alpha(\mathbf{x_i}) - L_\alpha(\mathbf{x_i})\Big)$.

Combining experts with ELM, the `FuseELM` approach: In the region, each station is associated with an expert [2]. We let $\mathcal{S}$ be the set of stations in the region and $|\mathcal{S}|$ its cardinality. In the region, each expert observes the current day solar PV production and local meteorological inputs. We apply the PI based ELM to each expert. This means that $B_M + B_N$ ELMs should need to be run for each expert. However, the information coming from each expert should be weighted by the trust granted to this expert's report. We introduce a loss vector, defined in each station and in each training data as the square of the difference between the target and the station forecaster: $\mathcal{L}_{s,i} \triangleq \Big(t_i - \hat{y}_s(\mathbf{x_i})\Big)^2, \forall i = 1, ..., \text{size}_{Train}, \forall s \in \mathcal{S}$. We let: $w_{s,i} \triangleq w_{s,i}\Big(\hat{y}_s(\mathbf{x_i}), t_i, w_{s,i-1}\Big)$ be the weight associated with the s-th expert in the region evaluated in the i-th training data. For each station s in the region, we update the exponentially weighted average forecaster over each training data i according to the rule: $w_{s,i} = \frac{w_{s,i-1}\exp(-\eta\mathcal{L}_{s,i})}{\sum_{s'=1,...,|\mathcal{S}|} w_{s',i-1}\exp(-\eta\mathcal{L}_{s',i})}$ with $\eta = \sqrt{2\ln(|\mathcal{S}|)/\text{size}_{Train}}$ [2]. An expert is said to be activated if its weight is large enough to guarantee its selection in the PI estimation at the regional scale. Once the final weight vector $(w_{s,\text{size}_{Train}})_{s\in\mathcal{S}}$

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8. Available from http://www.i6doc.com/en/.

is obtained, we order the experts according to decreasing weights and keep the index of the experts so that the sum of the activated experts weights equals the desired confidence level, starting with the expert having the highest weight. Then, the regional scale PI is obtained based on the activated experts PIs.

3 Experiments

Data description: The total half hourly solar production for the year 2013 is collected from the French transmission system operator[2]. These values are divided by the corresponding installed capacity to obtain a consistent PV production ratio. An approximate center is determined by inspection for the PACA region. The clear sky solar radiation data are collected from the SoDa service[3] and the sun position in the sky is computed using the SPA algorithm, with half hour time granularity.

Measurements of weather parameters at meteorological stations are collected from the Météociel platform[4]. The collected meteorological series include measurements of wind speed, wind direction, cloud cover, temperature, atmospheric pressure, relative humidity and precipitation at 13 stations, with hourly resolution for the year 2013. As many of these series have missing data for one or many meteorological inputs, series with more than 10% of missing data are removed from the database. The meteorological series left have, in average, 0.98% of missing data.

Performance: We start by running the ELM algorithm at a short-term horizon (e.g., one hour) taking as input the solar PV production of the current hour to predict the solar PV production of the next hour. The hidden node parameters are distributed according to a Gaussian density function. The PIs with nominal confidence 90% over one week in Spring 2013 is displayed in Figure 1 (a). PICP and Score corresponding to this reliability level are 0.940 and 0.068 respectively; which is extremely good in terms of both reliability and sharpness. However, performance measurements over longer term horizons (such as 24 hours) are significantly worse. This can be partly explained by the fact that the data contain many missing values and by the presence of outliers. Furthermore, basic ELM fails to capture extreme events such as unexpectedly high/low solar PV productions over such a long-term horizon.

So, we turn to the `FuseELM` algorithm, to which we add additional input variables. However, there is a tradeoff to be found between adding "informative" input variables which will increase the forecaster overall performance and adding data whose quality will alter the forecaster performance introducing undesirable biases. The influence of the various meteorological inputs known as having a non trivial impact on solar production forecasting is evaluated as a function of the algorithm performance measurements in Table 1. The performance (PICPs and Scores) are computed over one week in Spring 2013. In the evaluation process,

[2] Eco2mix http://www.rte-france.com/fr/eco2mix/eco2mix/
[3] SoDa http://www.soda-is.com/eng/services/services_radiation_free_eng.php
[4] Météociel http://meteociel.fr/obs/pays.php.

ESANN 2015 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 22-24 April 2015, i6doc.com publ., ISBN 978-287587014-8. Available from http://www.i6doc.com/en/.

Variables	PICP	Score	Nb of Experts	Experts Index
Wind Speed	0.904	0.093	8/13	4, 6, 12, 1, 11, 8, 3, 9
Wind Dir.	0.889	0.095	8/13	4, 12, 6, 1, 8, 11, 3, 9
Cloud Cover	0.910	0.094	8/13	4, 12, 6, 1, 11, 8, 3, 9
Precipitation	0.916	0.098	8/13	4, 6, 1, 12, 11, 8, 3, 9
Temperature	0.907	0.098	8/13	4, 12, 6, 1, 11, 8, 3, 9
Pressure	0.875	0.090	8/13	4, 6, 12, 1, 11, 8, 3, 9
Sun Zenith	0.886	0.091	8/13	4, 6, 12, 1, 8, 11, 3, 9
Sun Azimuth	0.913	0.097	8/13	4, 12, 6, 1, 8, 11, 3, 9

Table 1: `FuseELM` performance measurements depending on meteorological inputs for a confidence level of 90% in Spring 2013.

we give a higher priority to the reliability captured by the PICPs since it is the key feature reflecting the correctness of the constructed PIs [7]. We check according to Table 1 that precipitation, followed by sun position and cloud cover, is the indicator with the greatest influence.

We now run `FuseELM` taking as input, apart from the daily solar PV power production, daily precipitation. The PIs with nominal confidence 90% and the measured solar PV production in Spring 2013 obtained by the proposed `FuseELM` approach are plotted in Figure 1 (*b*). The optimal number of hidden nodes for the ELM is determined based on cross-validation. In Figure 1 (*d*), we have represented the Root Mean Square Error (RMSE) and Mean Average Error (MAE) as functions of the number of ELM hidden neurons. We check that 50 hidden neurons guarantee optimal learning capabilities while avoiding overtraining. In Table 1, we observe that 8 over 13 experts are activated. The indices of the experts are listed following decreasing weights in the probability vector $(w_{s,\text{size}_{Train}})_{s\in\mathcal{S}}$. The 90% confidence PIs obtained by the proposed `FuseELM` and the measured solar PV production in Winter 2013 is displayed in Figure 1 (c). The performance for Winter and Fall are worse than for Spring and Summer. This is due to the chaotic behavior of the solar PV production for some days of the week and the presence of extreme (unexpectedly low) production values.

Updating the experts weight by solving a constrained linear system [4] leads to PICP= 0.914 and Score= 0.101 as performance measures, with 5/13 experts being activated (indices $4, 8, 6, 12, 11$). Therefore, it is outperformed by `FuseELM`.

4 Conclusion

The learning algorithm proposed in this article combines distributed ELMs and information fusion to produce a 24 hour horizon probabilistic forecast of solar PV power production using daily solar PV power production and local meteorological inputs. Experiments run over one region in France, Provence-Alpes-Côte d'Azur, show that the most valuable model input, apart from the present day solar PV production, is precipitation. An interesting perspective would be to theoretically compare experts advice fusion with the weighted averaging techniques of varying degrees of complexity already widespread in information fusion [3].

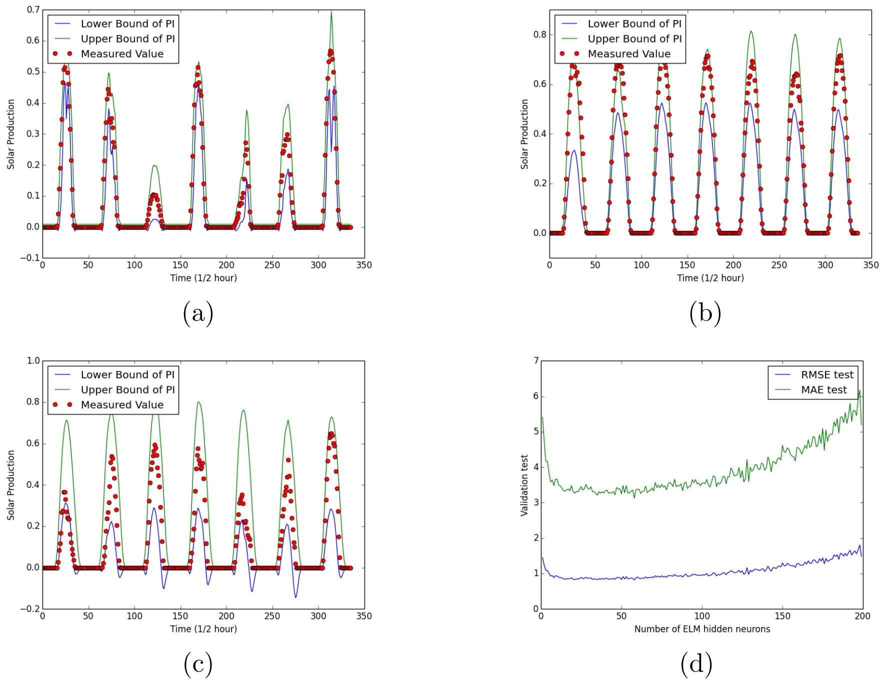

Fig. 1: PIs with nominal confidence 90%: hourly forecast (a) and daily forecast over one week in Spring (b) and Winter 2013 (c). Cross-validation test for ELM with different number of hidden neurons (d).

References

[1] G.-B. Huang, An Insight into Extreme Learning Machines: Random Neurons, Random Features and Kernels, *Cognitive Computation*, 6:376-390, 2014.

[2] H. Le Cadre and J.-S. Bedo, Distributed Learning in Hierarchical Networks, *proceedings of the* 6^{th} *International (ACM) Conference on Performance Evaluation and Tools* (*ValueTools*), pages $188-197$, October $9-12$, 2012.

[3] J.-P. Le Cadre, V. Nimier and R. Reynaud, Fusion en Traitement du Signal. In I. Bloch, editor, *Fusion d'informations en traitement du signal et des images*, Chapter 2, pages 29-52, Hermès, 2003.

[4] Y. Miche, E. Eirola, P. Bas, O. Simula, C. Jutten, A. Lendasse and M. Verleysen, Ensemble Modeling with a Constrained Linear System of Leave-One-Out Outputs, *proceedings of the European Symposium on Artificial Neural Networks*, d-side pub., April 28-30, Bruges (Belgium), 2010.

[5] A. Papavasiliou and S. S. Oren, Multiarea stochastic unit commitment for high wind penetration in a transmission constrained network, *Operations Research*, $61:578-592$, 2013.

[6] I. Reda and A. Andreas, Solar Position Algorithm for Solar Radiation Applications, *NREL Report No. TP-560-34302*, 55 pp., 2003, revised January 2008.

[7] C. Wan, Z. Xu, P. Pinson, Z. Y. Dong and K. P. Wong, Probabilistic Forecasting of Wind Power Generation Using Extreme Learning Machine, *IEEE Transactions on Power Systems*, $29:1033-1044$, 2014.