Le 3ème Séminaire International sur les Energies Nouvelles et Renouvelables
The 3nd International Seminar on New and Renewable Energies
Unité de Recherche Appliquée en Energies Renouvelables,
Ghardaïa – Algérie 13 et 14 Octobre 2014

Forecasting the produced power of a 20 Kwp grid-connected photovoltaic plant using SVM and MLP

Moufida Bouzerdoum[#1], Adel Mellit[#2]

[#]*Faculty of Sciences Engineering, Renewable Energy Laboratory, Jijel University, Algeria*

[1]m83.bouzerdoum@yahoo.com

[2]a.mellit@ yahoo.co.uk

***Abstract*— This paper presents a comparative study between support vector machine (SVM) and Multi-layer perceptron (MLP) for predicting the power output of PV systems. An experimental database collected from a 20 kWp grid-connected photovoltaic plant was used. Results confirm the ability of the techniques to forecast the produced power, however, SVM outperforms the MLP in the point of view accuracy.**

***Keywords*— Photovoltaic systems, forecasting, power output, Support Vector Machine (SVM), Multi-Layer Perceptron (MLP).**

I. Introduction

As reported by the IEA, global photovoltaic capacity has been increasing at an average annual growth rate of more than 40% since 2000 and it has significant potential for long-term growth over the next decades. By 2050, PV will provide 11% of global electricity production (4500 TWh per year), corresponding to 3000 GW of cumulative installed PV capacity.

Forecast the power produced by a PV plant is useful for:

- checking the performance of the plant;
- detecting anomalies and faults in the plant;
- making dispatching plans for the grid operators;
- helping operation and maintenance operators in choosing the most suitable timing for off-grid maintenance.

During the last decade, statistical learning algorithms have attracted much interest in academia and in companies of various industries. They have been successfully implemented for the performance of tasks related to predictive statistical process observed for which we can identify several variables.

This study focuses on two particular classes of these algorithms: artificial neural networks (multilayer perception: MLP) type and support vector machines for regression problems (SVMr) to forecast the produced output power of a grid-connected PV plant as an application in the domain of photovoltaic solar energy.

This paper is organized as follows: brief introduction on SVM and MLP is given in Section II. Database description is provided in Section III. Results and discussion are presented in Section IV.

II. Support Vector Machine and Multilayer Percptron Networks

A. Support vector machine for regression (SVMr)

The support vector machines are statistics learning tools introduced by Vapnik [1] in 1995. SVM are usually used in classification problems. The approach allows defining complex surfaces in spaces of large dimensions, with very concise representations. If traditional methods of learning based on the minimization of the training error (empirical risk), the main advantage of SVM is the possibility of determining an error (the risk structure) valid for validation. Based on the principle of SVM, the support Vector Regression (SVR) can treat problems of regression (linear or nonlinear). Recently several studies have been devoted to the use of SVR for function approximation and time series prediction [2, 3, 4].

The basic idea of SVR prediction is described as follows: Suppose we are given training data $\{(x_i, y_i)\}(i = 1,2, \dots, n)$, where each $x_i \in R^n$ is the input vector with n dimension, $y_i \in R$ is the associated desired output value of x_i. The SVR algorithm is as follows [1, 5]: Given a positive real number ε, we find a function f such that:

- $f(x_i)$ returns a value that does not deviate of y_i more than ε ;
- f is "simple" (flat) as possible.

To find such a function, the SVR algorithm determines f as a linear function of the form:

$$f(x) = \langle w, \varphi(x) \rangle + b \qquad (1)$$

Where $\varphi(x)$ is called the feature that is nonlinearly mapping from the input space x. w is the vector of the parameters (or weights) and b is a constant to be determined. To ensure the flatness of the function f, the standard weight $\|w\|$ is minimized. So the problem is to minimize this standard by ensuring that errors are less than ε and can be written

$$\min \frac{1}{2}\|w\|^2$$
$$\text{s.t } |y_i - \langle w, \varphi(x_i) \rangle - b| \leq \varepsilon, i = 1, \dots, N \qquad (2)$$

This description of the problem considers a linear function *f* that approximates all the examples with accuracy ε exists. In practice, this is not always the case. In the presence of excessive noise or outliers, it is more important to allow some errors. In this case, the concept of soft margin is used. It is to introduce slack variables ξ_i, ξ_i^* to make feasible the constraints of the optimization problem which becomes

$$\min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \quad (3)$$

under the constraints

$$y_i - \langle w, \varphi(x_i)\rangle - b \leq \varepsilon + \xi_i$$

$$\langle w, \varphi(x_i)\rangle + b - y_i \leq \varepsilon + \xi_i^*$$

$$\xi_i, \xi_i^* \geq 0, i = 1, \ldots, N \quad (4)$$

Where ξ_i and ξ_i^* respectively denote the positive and negative errors. The constant C > 0 is a hyper parameter be possible to adjust the compromise between the amount authorized error and the flatness of the function *f*. This formulation of the problem is to use an error function $|\xi|_\varepsilon$ called ε-insensitive of the form

$$|y - f(x)|_\varepsilon = \begin{cases} |y - f(x)| - \varepsilon, \text{for } |y - f(x)| > \varepsilon \\ 0 \qquad\qquad\quad , \text{otherwise} \end{cases} \quad (5)$$

The problem (4) is solved by minimizing the Lagrangian *L* function given by:

$$L = -\frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)k(x_i, x_j) - \varepsilon\sum_{i=1}^{N}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{N} y_i(\alpha_i - \alpha_i^*) \quad (6)$$

where α_i, α_i^* positives are Lagrange multipliers.
The weight of the model are determined by

$$w = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*)\varphi(x_i) \quad (7)$$

and the model can be written as

$$f(x) = \sum_{i=1}^{N} y_i(\alpha_i - \alpha_i^*)k(x_i, x) + b \quad (8)$$

The bias parameter *b* can be calculated by the conditions of Karush-Kuhn-Tucker(KKT).
The function *k* is called kernel function. The most commonly used for SVM kernels are polynomial kernels, sigmoidal and radial basis function (RBF) defined as follows:

- Linear : $k(x, x') = \langle x, x'\rangle$;
- Polynomial : $k(x, x') = (\gamma\langle x, x'\rangle + c)^d$;
- Sigmoidal : $k(x, x') = tanh(\gamma\langle x, x'\rangle + c)$;
- RBF : $k(x, x') = exp\left(\frac{-\|x, x'\|^2}{2\sigma^2}\right)$

Her γ , *c* and *d* are kernel parameters

B. Multi-layer perceptron (MLP)

The Presented first by Y. Le Cun [6], Multi Layer Perceptron (MLP) neural networks are feed forward neural networks type, consisting of one or more hidden layers and an output layer. Each layer of the network is composed of artificial neurons. The first hidden layer receives information from the inputs. The information is processed and sent to the following layers until the last. MLPs are known as universal approximates and are used in nonlinear regression problems. Neurons are connected together by weighted connections. The weights of those connections are that govern the operation of the network and program an application from the input space to the output space through a non-linear transformation. The creation of a multilayer perception to solve a given problem therefore involves inferring the best possible application as defined by a set of training data consisting of pairs of input vectors and desired outputs. This inference can be, among other things, by the algorithm called back propagation.

For a 2-layer neural network with *N* inputs *h* hidden neurons, the relationship between the input *x* and output *y* is given by:

$$y = y(x, w) = g\{b_0 + \sum_{j=1}^{h}[w_{0j} f(b_{jh} + \sum_{i=1}^{N} w_{ji}.x_i \quad (9)$$

where b_0 is the bias at the output layer, w_{0j} is the weight connection between neuron *j* of the hidden layer and the single output neuron, b_{jh} is the bias at neuron *j* of the hidden layer (*j*=1, *h*), w_{ji} is the weight connection between input variable (*i*=1, *N*) and neuron *j* of the hidden layer, x_i is the input parameter *i* , and the function $g(\cdot)$ is the nonlinear transfer function (also called activation function) at the output node and $f(\cdot)$ is the common nonlinear transfer function at each of the hidden nodes. The activation function *f* is usually taken to be sigmoid, and therefore nonlinear, the most common choices being the log-sigmoid, and the tan-sigmoid. The training process of an MLP consists of determining the optimal set of weights, and in some cases the network's structure, which minimizes a measure of error. In this case the Levenberg-Marquardt (LM) algorithm [7] has been applied, since it has been reported to avoid local minima better than other classical training algorithm such as the Back Propagation approach.

III. DATABASE DESCRIPTION

The used power output data have been collected from a 20 kWp GCPV plant installed on the roof top of the municipality of Trieste, Italy. We have used data from 29 January to 25 May 2009 to evaluate the performance of the forecasting models (a sample is taken every 10 min). As example figure 1 shows the evolution of the produced output power versus time.

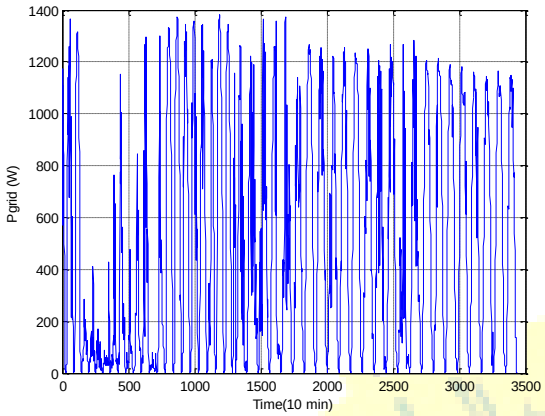

Fig.1 Power output data

A set of 3437 samples has been divided into two subsets: 89,5% (3079 patterns) of the samples of the total set have been used for the training of the networks and SVM, while the remaining 10,5% (358 patterns) have been used for testing and validation.

IV. Results and Discussion

Before applying the training algorithm, the following preprocessing, which makes the network more efficient, has been carried out on the datasets [8, 9]:

$$y_i^* = \frac{y_i - y_{min}}{y_{max} - y_{min}} \quad (10)$$

Where y_i is the original data value, y_i^* is the corresponding normalized variable, y_{min} is the minimum values in $\{y_i\}$, y_{max} is the maximum value in $\{y_i\}$.

The procedure applied in the development of the MLP network and SVMr model was as follows: first, data inputs are normalized as described before, and then training, test and validation sets were selected. After that, the parameters were chosen to create and train the models. Finally, the data were unormalized and the performance of the models was checked based on the error between the outputs values and the inputs ones.

The implementation of MLP problem requires selecting the training algorithm, the activation function, the normalization procedure, the number of hidden layers and the numbers of nodes in the hidden layers.

Different architectures of MLP have been evaluated, and the best one has been chosen consisting of one hidden layer within 16neurons as shown in figure 2. The Levenberg-Marquardt (LM) has been employed and the activation function used was the log-sigmoid function. After 1000 iterations, the developed MLP model converges to the fixed error.

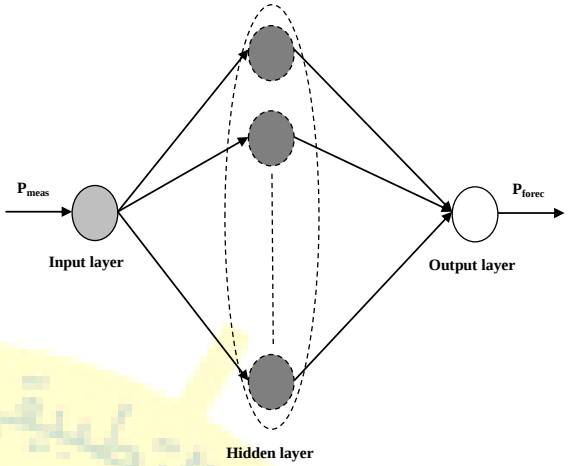

Fig.2 The used MLP architecture

Different factors affect the accuracy of the SVM problem and therefore must be chosen carefully. These are the kernel function and the kernel parameter (γ), ε-insensitive loss and the upper bound C. The optimum parameters should be chosen.

Previous studies has shown that Gaussian Radial Basis Function (RBF) gives better results in time series predictions than other parameters function [4,10].The kernel SVM have a great affection the accuracy of the prediction. Different combinations of C, γ and ε were examined and the best combination of performance was selected. The values are determined by their tests as follows: C =0.0024, γ=0.021and ε=0.93.The optimum parameters were selected based on the lowest error in the validation step and makes SVM have high generalization.

The main contribution in this study is to compare the produced and the forecasted output power using MLP and SVM models. Results are reported in figure 3. As can be seen, both designed techniques achieved high precision.

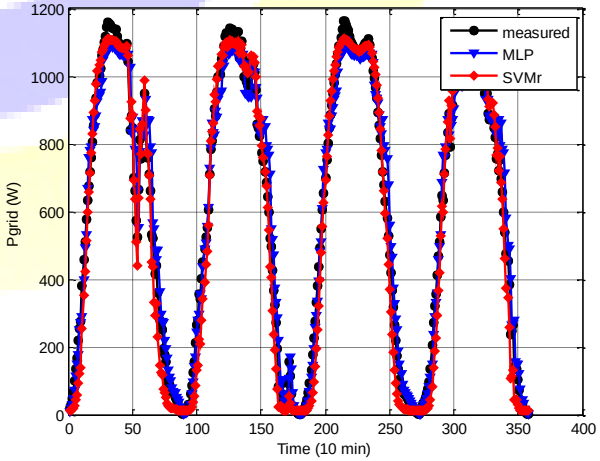

Fig.3 Data of power output and its forecasts by MLP and SVMr

Figure 4 shows the forecasted power using MLP and SVM versus measured values of the produced power produced by GCPV plant during the period evaluated (from 21th to 24th May). A very good performance of both techniques are observed as the correlation coefficient is greater than 98%.

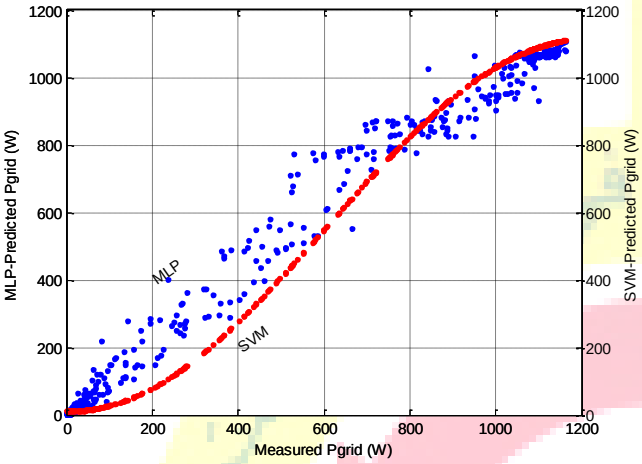

Fig.4 Data of power output and its forecasts by MLP and SVMr

In order to evaluate the performance of the model which allows us to estimate and predict the power output, statistical tests, which are common for all types of prediction [11, 12] such as the NRMSE (Normal Root Mean Square Error), the NMBE (Normalized Mean Bias Error) and MPE (Mean Percentage Error) are calculated to compare the measured values to those calculated using the model.

$$MPE = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{P_{i,meas}-P_{i,forec}}{P_{i,meas}}\right) * 100 \quad (11)$$

$$MBE = \frac{1}{N}\sum_{i=1}^{N}\left(P_{i,meas} - P_{i,forec}\right) \quad (12)$$

$$NMBE = \frac{MBE}{\frac{1}{N}\sum_{i=1}^{N} P_{i,pred}} * 100 \quad (13)$$

$$RMSE = \left[\frac{1}{N}\sum_{i=1}^{N}\left(P_{i,meas} - P_{i,forec}\right)^2\right]^{1/2} \quad (14)$$

$$NMRSE = \frac{RMSE}{\frac{1}{N}\sum_{i=1}^{N} P_{i,forec}} * 100 \quad (15)$$

Where $P_{i,meas}$ and $P_{i,forec}$, are respectively the measured and forecasted values of the power output at time i and N is the number of measured values (calculated).

Table 1 shows the values of the errors of the forecasts obtained with the designed models developed to predict the power output.

TABLE I
STATISTICAL TEST FOR POWER OUTPUT

	RMSE (W)	NRMSE %	MBE	NMBE %	MPE %	R
SVMr	59.9541	9.8515	-28.9395	-4.7553	-1.2197	0.9946
MLP	63.3062	10.4023	5.8068	0.9542	8.7130	0.9888

With respect to Table 1, it can be seen that the developed MLP model achieved high precision but the SVMr performs better than MLP. The RMSE of the forecast carried out with SVMr was more than 3W lower than the one obtained by the MLP. Moreover, both the MBE and MPE of the forecasts of power production carried out with SVMr were also significantly lower than the respective errors of the forecasts carried out with MLP. Its overall the forecasting accuracy is improved by the SVMr model.

V. CONCLUSION

In this work, a support vector machines for regression problem (SVMr) and a multi-Layer perceptron (MLP) has been successfully developed for forecasting (10min ahead) the power output produced by a 20 kWp GCPV system. A comparison based on various statistical tests such as RMSE, MBE, and the correlation coefficient have shown that the SVMr performs better than MLP. In order to improve the results, future research will focus on the use of hybrid models based on SVM technique and times series models.

ACKNOWLEDGMENT

Authors would like to thank Dr. A. Massi Pavan (Trieste Univ., Italy), for the used database. This work was partially supported by the TWAS under grant Ref. 09-108 RG/REN/AF/AC_C: UNESCO FR: 3240231224.

REFERENCES

[1] V.Vapnik, *The nature of statistical learning theory*, New York: Springer-Verlag, NY, USA, 1995.
[2] J. Shena, Y.R. Syaub, and E.S. Lee, "Support vector fuzzy adaptive network in regression analysis", *Computers and Mathematics with Applications*, vol. 54, no.11-12, pp.1353-1366, Dec. 2007.
[3] B.R. Chang and H.F. Tsai, "Forecast approach using neural network adaptation to support vector regression grey model and generalized auto-regressive conditional heteroscedasticity", *Expert Systems with Applications*, vol. 34, no. 2, pp.925-934, Feb. 2008.
[4] Z. Xin and C.Tian-Lun, "Nonlinear time series forecast using radial basis function neural networks", *Commun. Theor. Phys.*,vol. 40, no. 2, pp. 165-168, Aug. 2003.
[5] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression", *Statistics and Computing*, vol.14, no. 3, pp.199-222, Aug. 2004.
[6] Y. Le Cun,"A learning scheme for threshold networks", *In Cognitiva'95*, pp.599-604, France, 1985.
[7] MT. Hagan and MB. Menhaj, "Training feed forward network with the Marquardt algorithm",*IEEE Transactions on Neural Networks*,vol.5,n°6, pp. 989-993, 1994.

[8] C. Chen and S. Duan, T. Cai, B. Liu, "Online 24-h solar power forecasting based on weather type classification using artificial neural network", *Solar Energy*, vol.85, no.11, pp. 2856-2870, Nov. 2011.

[9] S. Jhi, W. Lee, Y. Liu, Y.Yang, and P.Wang, "Forecasting power output of photovoltaic systems based on weather classification and support vector machines", *IEEE transactions on industry applications*, vol. 48, no. 3, pp. 1064-1069, May/Jun. 2012.

[10] P. Yee and S. Haykin, "A dynamic regularized radial basis function network for nonlinear, nonstationary time series prediction", *IEEE Transactions on signal processing*, vol. 47, no.9, Sep.1999.

[11] S. A. Kalogirou, "Artificial neural networks in renewable energy systems applications: a review". *Renewable and Sustainable Energy Reviews*, vol.5, no.4, pp. 373-401, Jun. 2001.

[12] A. Mellit and S. A. Kalogirou, "Artificial intelligence techniques for photovoltaic applications: A review", *Progress in Energy and Combustion Science*, vol.1, no.1, pp. 52-76, Nov. 2008.