Beyond Mean Square Error: training mappers with Entropy for Wind Power prediction

V. Miranda, *Fellow IEEE*, C. Cerqueira and C. Monteiro

***Abstract* — This paper shows the error currently being committed when Mean Square Error is used as the optimizing criterion in training neural networks or fuzzy inference systems. The illustrative case is wind park power output prediction, based on wind speed and direction. It presents the application of Renyi's Entropy combined with Parzen windows as a measure of information content of the error distribution in model parameter estimation in supervised learning. It shows that in the prediction of power generated in a wind park, made by Takagi-Sugeno Fuzzy Inference Systems, whose parameters are discovered with an EPSO – Evolutionary Particle Swarm Optimization algorithm, an entropy criterion leads to narrower error distribution functions than MSE.**

***Index Terms*—Information theoretic learning, entropy, fuzzy inference systems, wind power generation.**

I. Objective

This paper discusses the advantage of generating a prediction for wind power generation for a wind park based on wind speed and direction information using, as training criterion, not the usual MSE – Mean Square Error, but a measure of Information Entropy.

The prediction of power output from a wind park is highly important presently in Europe, where the growing penetration of wind generation will reach heavy percentages (in the range of 5 to 20%) in some countries in the coming years, like Germany, Spain, Denmark or Portugal, because of the collective effort in the European Union in complying with the Kyoto protocol. For instance, in Portugal by 2010 some 5100 MW of wind generators will be installed, relating to a country peak power consumption in 2006 of about 8500 MW. So, we are no longer talking of marginal effects.

The prediction of the available wind power influences the market clearing price, because wind power must be placed as an offer at zero price before the matching of offer and demand. Errors in wind prediction will distort market prices and, therefore, accurate and unbiased predictions are of the utmost importance. Also, a TSO (Transmission System Operator) will need the best wind power prediction possible to economically plan the operation of the network, namely for the definition of contracts correcting the power balance.

V. Miranda is with INESC Porto and also with FEUP, Faculty of Engineering of the University of Porto, Portugal (e-mail: vmiranda@inescporto.pt). INESC Porto is a private, non for profit, research institute with one area devoted to power systems (http://power.inescporto.pt).

Cristina Cerqueira is with INESC Porto and also with FCUP, Faculty of Sciences of the University of Porto, Portugal (email: caac@inescporto.pt).

Claudio Monteiro is is with INESC Porto and also with FEUP (email: cmonteiro@inescporto.pt)

The basic data for short term wind power prediction (up to six hours) are wind speed and wind direction. It isn't feasible neither to collect on line data at the place of each turbine (some 4000 turbines will be soon operating in Portugal) nor to model local effects representing individually each turbine and its installation condition and terrain effects. Presently, data are collected at a measuring station in the neighborhood of the park – but some older parks don't even have a reliable data collecting station and wind data must be predicted by other methods that may take in account meso-scale effects or meteorological data. From these data one must predict the actual wind power production from the aggregate of tens or hundreds of generators geographically spread in a region. This task is further complicated by a number of factors such as:

a) Wind speed may vary from generator to generator – and wind direction too, in a certain extent
b) Tail or shadow effects that reduce the energy of the wind behind a turbine become more or less important depending on the layout of the park and on the direction of wind.
c) A complex terrain produces unexpected effects.
d) The non-linear characteristic of the curve of power vs. wind speed of generators adds further complexity to the problem.

The prediction of power output from a wind park is a necessary phase in methods of wind forecasting that rely on a wind forecast as an intermediate step.

Prediktor[1], for instance, derives from a model of fluid dynamics equations, and converts it to wind as seen by a wind park, but then derives power from theoretical power curves, which is a weak step. Also *eWind*[2] forecasts first wind speed and only then wind power.

This two-step approach may be found also in pure statistical methods [3] and in methods based on computational intelligence techniques [4]. When the terrain is complex, sophisticated models such as *VENTOS* [5] are used to describe air flows, especially in the wind park planning phase, but not so much in the operation of the power system.

Some methods convert wind predictions into wind power predictions by using an empirical *power curve* that tries to represent the non-linear behavior of wind generators. Lange [6], for instance, derived a model for uncertainties in power prediction by relating the standard deviation of their errors with the standard deviation of errors in wind predictions and the local slope of such power curve. However, the input-output relation between wind and power is much more complex than represented by a single non-linear function,

which introduces unnecessary noise. We believe that mapping methods such as neural networks or fuzzy inference systems are better suited to emulate such relation.

This paper follows this line of reasoning. It is about training a mapper for emulating the input-output transfer function relating wind speed/direction to power output of a wind park. But it is most of all about training *better* mappers, leading to more accurate output results: leading to a narrower probability distribution function of the prediction errors.

The most widely cost function used in mapper training is the MSE – Mean Square Error. It is embedded in most backpropagation software applications to train neural networks or fuzzy inference systems. Its use has become so automatic that most researchers, namely in the Power Systems community, take it for granted.

The use of MSE has the implicit assumption that errors are Gaussian distributed. This is because Gaussian distributions are the only ones that have all information contained in their first two moments (mean and variance), and minimizing the square error is the same as minimizing variance. However, this assumption does not hold in many problems. Errors in wind park power output predictions are sometimes far from being Gaussian. Even if wind predictions are produced with Gaussian errors, the non-linearity of the characteristic curve of wind turbines causes power predictions to display non-Gaussian characteristics. This has been shown, for instance, in [7], for 20 sites in Germany over a period of 3 years. Typically, error distributions from wind power prediction models are right skewed and have positive excess of kurtosis, meaning that: they are asymmetrical, they present a higher frequency of errors to the left of the mean and are flatter than the Gaussian distribution.

When we apply a MSE criterion to train a mapper, we pass to its parameters just a fraction of the information contained in the input set and leave useful information in the error distribution, residing in its higher moments which are not used in the optimization. This is the basic flaw incurred in by all those that adopt MSE to train a mapper.

To avoid this problem and use all information possible in data, leaving the error distribution with as little information as possible, on has to use an optimization criterion that takes in account all moments of output distribution – such as an Entropy criterion. This paper is devoted to showing that Information Theoretical Learning (ITL) concepts, which base learning on Renyi's Entropy measure, are an adequate way to deal with non Gaussian error distributions. We will show that training a Fuzzy Inference System (FIS) of the Takagi-Sugeno type, using an Entropy criterion, leads to prediction error distributions that are narrower than using what "everybody" uses, the MSE criterion.

II. Training Mappers

Any system that emulates an input-output transfer function and whose performance depends on the tuning of internal weights or parameters is called *a mapper*. Neural networks or fuzzy inference systems are mappers, for instance, but so are time series generated by ARIMA methods, for instance.

To train a mapper one must consider a three block structure, such as depicted in Figure 1:

a) The mapper **g**, with its internal structure and weights **w**
b) The performance criterion.
c) The training algorithm.

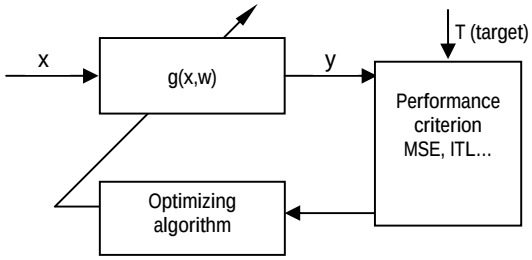

Figure 1 – Basic arrangement of a mapper identifying its three main modules

The basic idea for a supervised training of a mapper with an Entropy criterion is to select a set of weights that will lead to an output presenting a distribution of (Target-Output) errors as a Dirac function (meaning that all errors would be equal, see Figure 2). And just by adding to the results a bias corresponding to the mean of the pdf of the errors, i.e., the deviation from zero, we will have reached a machine whose output exactly reproduces the real data.

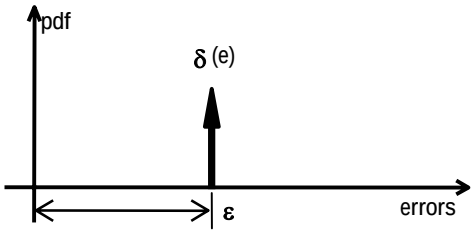

Figure 2– A mapper producing a systematic error ε for all inputs will display a error density function like a Dirac function

A Dirac function has minimum entropy. Therefore, it is easy to understand that training a mapper to minimize the entropy of the pdf (probability distribution function) of the errors approximates us to the Dirac function and minimizes the information content of that distribution.

III. Entropy

Information Theoretic Learning (ITL) is a framework that became successful in developing an approach to estimating the entropy of a set given a certain sample [8][9]. Entropy is a concept developed in information theory that formalizes the notion of information content. The less predictable a message is, the larger is its information content; a message perfectly know *a priori* has a zero information content.

Shannon [10] defined the entropy of a probability distribution $P = (p_1, p_2, \ldots, p_n)$ as

$$H_S(P) = \sum_{k=1}^{N} p_k \log \frac{1}{p_k} \quad \text{with} \quad \sum_{k=1}^{N} p_k \quad \text{and} \quad p_k \geq 0 \qquad (1)$$

Although this definition has been widely applied, namely in communication systems, other definitions are possible. Renyi's entropy [11] is defined as

$$H_{R\alpha} = \frac{1}{1-\alpha} \log \sum_{k=1}^{N} p_k^{\alpha} \quad \text{with} \quad \alpha > 0, \alpha \neq 1 \tag{2}$$

Renyi's entropy is a family of functions $H_{R\alpha}$ depending on a real parameter α. There is a relation between Shannon's and Renyi's definitions:

$$\begin{aligned} & H_{R\alpha} \geq H_S \geq H_{R\beta} \quad \text{if} \quad \beta > 1 > \alpha > 0 \\ & \lim_{\alpha \to 1} H_{R\alpha} = H_S \end{aligned} \tag{3}$$

When $\alpha = 2$, we have what is called quadratic entropy

$$H_{R2} = -\log \sum_{k=1}^{N} p_k^2 \tag{4}$$

This definition can be generalized for a continuous random variable Y with pdf $f_Y(z)$:

$$H_{R2} = -\log \int_{-\infty}^{+\infty} f_Y{}^2(z)dz \tag{5}$$

It can be shown that Shannon's entropy and Renyi's entropy inducer the same order relation in a functional space and therefore minimizing Reny'ys entropy leads to the same result as minimizing Renyi's entropy. However, Renyi's entropy, with its sum of probabilities, is much more amenable to algorithmic implementation than Shannon's entropy with its sum of weighted logarithms of probability.

IV. Estimating a pdf with Parzen windows

The estimation of the pdf of data from a sample constituted by discrete points $\mathbf{y}_i \in R^M$, i = 1,…,N in a M-dimensional space, may be done by the Parzen window method [12]. This technique uses a kernel function centered on each point; it looks at a point as being locally described by a probability density Dirac function, which is replaced or approximated by a continuous set whose density is represented by the kernel. If a Gaussian kernel is used, the expression of the estimation $\hat{f}_Y$ for the real pdf f_Y of a set of N points is a summation of individual contributions

$$\hat{f}_Y(\mathbf{z}) = \frac{1}{N} \sum_{i=1}^{N} G(\mathbf{z} - \mathbf{y}_i, \sigma^2 \mathbf{I}) \tag{6}$$

where G(.,.) is the Gaussian kernel and $\sigma^2\mathbf{I}$ is the covariance matrix (here assumed with independent and equal variances in all dimensions). In each dimension, we have

$$G(z_k - y_{ik}, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2\sigma^2}(z_k - y_{ik})^2} \tag{7}$$

It is easy to understand that the "size" of the window, here defined by the value of σ, is important in obtaining a smoother (for larger values) or more "spiky" estimate for f_Y. This is illustrated in Figure 3, for a sample of 8 points around which Gaussian windows have been placed.

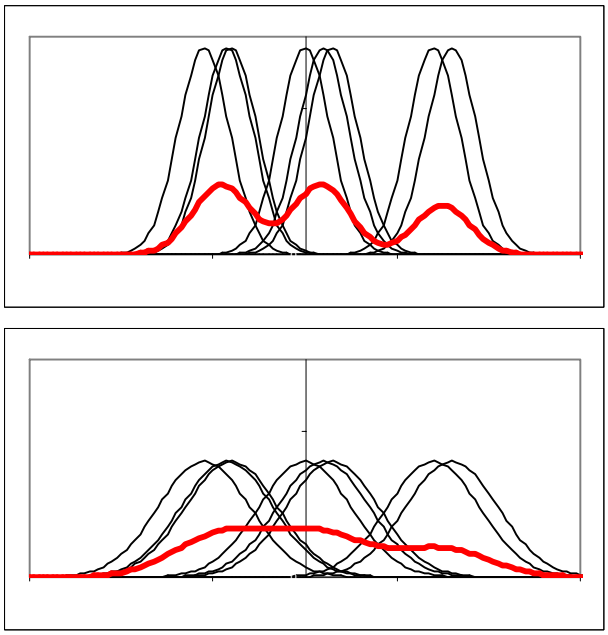

Figure 3 – Influence of the window size in the smoothness of the pdf estimation – narrower windows (above) lead to a more spiky approximation.

The ability to obtain well behaved approximations of the pdf of a set may be used with advantage to help the convergence of optimization algorithms that risk getting stuck in local optima of a more spiky function.

V. ITL criterion

Combining Renyi's definition of the entropy of a pdf with an estimate of the pdf by the Parzen window method, we reach an entropy estimator for a discrete set of data points $\{\mathbf{y}\}$ as

$$H_{R2}(\mathbf{y}) = -\log \int_{-\infty}^{+\infty} \hat{f}_Y{}^2(\mathbf{z})d\mathbf{z} = -\log V(\mathbf{y}) \tag{8}$$

$$V(\mathbf{y}) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \int_{-\infty}^{+\infty} G(\mathbf{z} - \mathbf{y}_i, \sigma^2 \mathbf{I}) G(\mathbf{z} - \mathbf{y}_i, \sigma^2 \mathbf{I}) d\mathbf{z} \tag{9}$$

In this expression we recognize the convolution of Gaussian functions, which has the following interesting result:

$$V(\mathbf{y}) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} G(\mathbf{y}_i - \mathbf{y}_j, 2\sigma^2 \mathbf{I}) \tag{10}$$

This means that, in order to calculate entropy, we do not have to calculate any integrals but simply the Gaussian function values of the vector distances between pairs of samples. In ITL vocabulary, V(**y**) is called the *information potential (IP)* of the data set. As the objective is to minimize H, one can instead maximize the information potential V. So, Max V becomes the cost function for optimizing a trainable mapper with minimum output entropy [13].

The discovery of weights in a mapper may be done by applying a suitable optimization method that will discover the

weights **w** that minimize the objective function

$$\min \; H_{R2}(\mathbf{w}) \tag{11}$$

This can be achieved by the classical back-propagation algorithm [14]. In this paper, however, we have applied an evolutionary algorithm to minimize entropy: EPSO, Evolutionary Particle Swarm Optimization.

One must stress that this is not speculative work. The theoretical foundations of ITL are solidly established and have been object of a considerable number of applications, from blind source separation [15] to clustering and to the supervised training of nonlinear adaptive systems [16]. However, to the authors' knowledge, it is the first time that such concepts are applied out of the context of signal processing and in the power system domain. This cross fertilization is hoped to be fruitful to the development of better models in power systems.

VI. Application to TS-FIS

The prediction of wind power output is done in this paper by a Takagi-Sugeno Fuzzy Inference System (TS-FIS) [17]. It may be viewed as neuro-fuzzy mapper and is commonly trained with the back-propagation algorithm under supervised training. The most widespread tool is the ANFIS [18]. One has training and test sets with target values **T** and the training task deals with the information content of the errors $\boldsymbol{\varepsilon} = \mathbf{T} - \mathbf{y}$ between target and the output **y**.

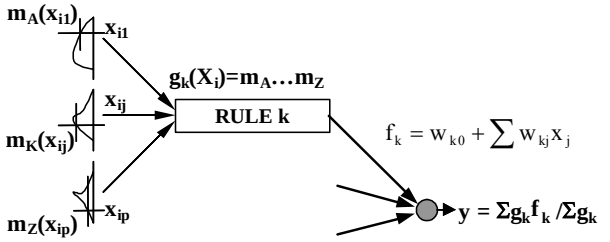

Figure 4 – TS-FIS scheme. Each input pattern X activates some membership functions; the combination of these fires a rule k with strength g_k. The weighted combination of rule firing strengths gives the output of the system.

In a TS FIS, one has rules that are fuzzy in their antecedent and crisp in their consequent. A general form of a rule k with output y_k is

IF (x_1 is A and … and x_p is Z) THEN $y_k = y(\mathbf{x}, \mathbf{w})$

The antecedent of rule k is a fuzzy set whose membership function g_k is the intersection of fuzzy sets describing conditions A,…,Z. Usually, the T-norm used to represent intersection is the product (of the membership values of each input variable).

The consequent of a rule k is a function f_k of inputs. In 0-order TS-FIS, f_k is constant and, therefore, $f_k = w_k$. In 1st-order TS-FIS, f_k is a linear combination of inputs such as in

$$f_k = w_k + w_{k1}x_1 + \ldots + w_{kp}x_p \tag{12}$$

The output of a TS-FIS is a weighted sum of the responses of all the rules

$$y_i = \sum_{k=1}^{R} \bar{g}_k f_k \;, \text{ with } \bar{g}_k = \frac{g_k}{\sum_{k=1}^{R} g_k} \tag{13}$$

To optimize the performance criterion, we have to discover the adequate weights **w** of the consequents of rules. Other parameters may be adjusted by training in TS-FIS. If the membership functions of the inputs are Gaussian functions, one can also calculate updates on central variance and spread of these functions. However, this is not convenient in many cases, because the inputs are associated with linguistic expressions and the change in the shape or location of the membership functions creates dissociation with the linguistic labels they are supposed to represent.

In this paper we are not applying ANFIS: it performs a minimization of the MSE. It must be highlighted that there are versions of ANFIS that add to the MSE function a term that has been called *information entropy* [18], because of the analogy with the formula of Shannon Entropy. However, this term is a function of the firing strengths of the FIS and not related to the data fitting. It has been introduced heuristically without any theoretical background to support it and does not have the effect of the optimization of the entropy of the error distribution as proposed by the ITL model.

VII. EPSO as the Optimizer

EPSO – Evolutionary Particle Swarm Optimization, is a hybrid in concepts of Evolutionary Algorithms and Particle Swarm Optimization [19][20] and with applications in Power Systems [21]. It is an Evolutionary Algorithm (close to the family of Evolution Strategies and Evolutionary Programming) where the recombination operator is made self-adaptive and is non-conventional: it is, in fact, the "movement rule" of PSO (Particle Swarm Optimization) methods.

Recombination is an operation that produces new offspring from some form of combination of parent individuals, chosen in the population (the classical recombination operator, in GA, is called crossover). The movement rule of PSO generates a new individual as a weighted combination of parents, which are: a given individual in the population, the best ancestor of this individual and the best ancestor of the present generation. This may be seen as a form of intermediary recombination. In this type of recombination in evolutionary algorithms, a new individual is formed from a weighted mix of ancestors, and this weighted mix may vary in each space dimension.

The recombination rule for EPSO is the following: given a particle $\mathbf{X}_i$, a new particle $\mathbf{X}_i^{new}$ results from

$$\mathbf{X}_i^{(k+1)} = \mathbf{X}_i^{(k)} + \mathbf{V}_i^{(k+1)} \tag{14}$$

$$\mathbf{V}_i^{(k+1)} = w_{i0}^{*}\mathbf{V}_i^{(k)} + w_{i1}^{*}(\mathbf{b}_i - \mathbf{X}_i) + Cw_{i2}^{*}(\mathbf{b}_g^{*} - \mathbf{X}_i) \tag{15}$$

where the symbol * indicates that these parameters will undergo evolution under a mutation process, and

$\mathbf{b}_i$ – best point found by the line of ancestors of individual i up to the current generation

$\mathbf{b}_g$ – best overall point found by the swarm of particle in their past life up to the current generation

$\mathbf{b}_g^* = \mathbf{b}_g + w_{i4}^* N(0,1)$ - it is an individual in the neighborhood of $\mathbf{b}_g$.

$\mathbf{X}_i^{(k)}$ – location of particle i at generation k

$\mathbf{V}_i^{(k)} = \mathbf{X}_i^{(k)} - \mathbf{X}_i^{(k-1)}$ – is the "velocity" of particle i at generation k

w_{i1} – weight of the *inertia* term (a new particle is created in the same direction as its previous couple of ancestors)

w_{i2} – weight of the *memory* term (the new particle is attracted to the best position occupied by its ancestors)

w_{i3} – weight of the *cooperation* or *information exchange* term (the new particle is attracted to the overall best-so-far found by the swarm).

w_{i4} – weight affecting dispersion around the best-so-far

C – a diagonal matrix with each element in the main diagonal being a binary variable equal to 1 with a given communication probability p and 0 with probability (1-p); in basic models, p = 1 but in advanced models p = 0.2 has proven to be more effective in assuring the progress of the algorithm, by limiting communication among the particles of the swarm – yet another means of shaping the recombination operator.

Assigning a value for p less than 1 implies the adoption of a communication topology among particles in the form of a *stochastic star* [22]. This constitutes an improvement over classical deterministic star topologies that allow EPSO algorithms to display better convergence properties.

EPSO is a self-adaptive algorithm because the weights that regulate recombination are taken as strategic parameters and are mutated and allowed to evolve. Selection acts on the recombination operator weights and, from generation to generation, a better (adaptive) recombination operator evolves.

In a diversity of problems, EPSO has been showing better performance than other meta-heuristics such as Genetic Algorithms or the classical Particle Swarm Optimization algorithm [23][24]. It tends to escape from local optima and is robust, i.e., generates results with a narrow variance in a series of runs for a problem with random initialization.

VIII. Prediction of Generation from a Wind Park

We will now present the results for the prediction of the power output of a real wind park located in northern Portugal, having a total installed capacity of about 40 MW with generators having each a capacity of about 1 MW located on the top of a mountain range.

The data for this exercise are composed of three time series: wind speed, wind direction and power output of the wind park, collected every ten minutes, collected from January 1, 2004 to February 20, 2005. For confidentiality reasons, actual power output has been transformed into a percentage of maximum available capacity of the park.

The objective is to show that the application of the ITL criterion may produce better mappers than the application of the classical Mean Square Error. To demonstrate this, we have trained two 0-order TS-FIS using an EPSO algorithm; the MSE model was meant to find weights that minimize the Mean Square Error, and the ITL model to find weights that minimize Renyi's Quadratic Entropy of the error distribution. We have selected 5000 points to train and test the FIS and divided them in a training set of 1000 points and a test set with the other points.

Figure 5 shows a plot of untreated data, as collected from the SCADA system, of 9993 measurements of wind speed vs. wind park power output. It is obvious that the representation of the wind speed/power output relation by a theoretical function will lead to a result far from satisfactory. The data used for training the TS-FIS have been subject to scrutiny to eliminate anomalous points – for instance, derived from the disconnection of generators for maintenance (that is why one may find in the figure some points with high wind speed and zero power output – but not to be confused with other points on the far right, where the zero output is explained by the cut-off security operation of the wind generators when wind speed is excessive).

In Figure 6 we plot the same data showing wind speed and direction, as measured at a point close to the wind park. It is quite obvious that wind frequency changes with wind direction. A plot like this helps in the design of the layout of wind parks.

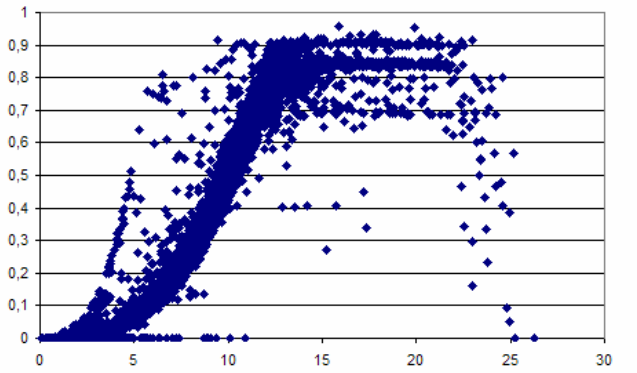

Figure 5 – Plot of wind speed (x axis) vs. power output of the wind park (y axis). Power output is represented in p.u. relative to the total installed capacity. Data untreated.

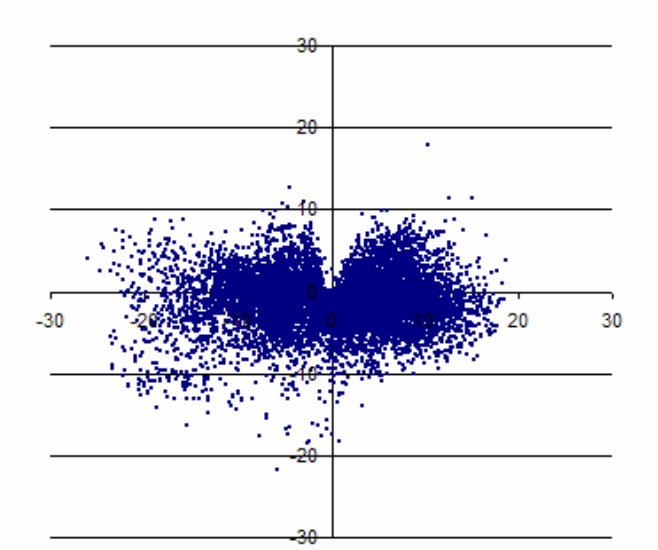

Figure 6 – Distribution of wind speed (in m/s) and direction at the measuring location near the wind park. Each point is the tip of a vector whose size is proportional to wind speed and angle is related to wind direction.

We have defined a 0-order TS-FIS, with the following characteristics:

a) Two input variables: wind speed S, in m/s, and wind direction D, in degrees
b) The range of S is between 0 and 30, and the range of D is between 0 and 360
c) The range of power output P is between 0 and 1
d) The universe of discourse of S was partitioned in 5 fuzzy sets with Gaussian membership function
e) The universe of discourse of D was partitioned in 2 fuzzy sets with Gaussian membership function

We have maintained these membership functions with central value and spread fixed and only optimized the weights **w** of the 10 fuzzy rules of the system. To find optimal weights, we used a simple EPSO algorithm with 20 individuals (particles) and replication factor r = 2 (each parent gives birth to two descendants). We used Gaussian mutations with learning rate $\tau = 0.5$ and communication probability p=0.2.

We trained two models with EPSO: minimizing MSE and Entropy. The same stopping criterion was used for both models (number of iterations without improvement of the fitness function) and they only differed in the fitness function used. For the Entropy model, we the results were obtained using Gaussian Parzen windows with fixed size ($\sigma = 0.01$).

From the theory behind the models, and knowing that the errors cannot have a Gaussian distributions (the non-linearity evident in Figure 5 makes sure of it), we expected that the error distribution of the predictions produced by the model trained with the Entropy criterion would be narrower than the one produced by the model trained under the MSE criterion.

Figure 7 and Figure 8 plot the probability density functions of the prediction errors for both models, in the training and in the test set. In the case of the Entropy model, we have already subtracted the adequate bias value so that the mean error value became zero in both distributions. They clearly show that the error distributions from predictions of the MSE model have fewer values close to 0 than the distribution of errors resulting from the ITL model. This was an expected result [25].

This means that the entropy model generates an error pdf closer to a Dirac function. It is a better result from any point of view than the one produced by the MSE model. This is seen as a valuable and meaningful result because it demonstrates in a practical real world example what the ITL theory was predicting, i.e., that using an Entropy criterion would lead to a prediction model with errors in general closer to zero than with a system trained using only variance (the MSE) as the optimizing criterion.

The value of Renyi's Entropy, for the error distribution in the test set, generated by the application of the MSE model, was of -0.6648, while with the application of the Entropy model it was of -1.1782 – naturally, a smaller value.

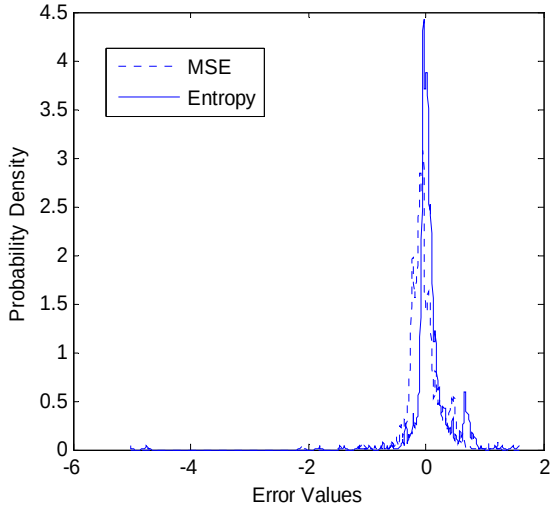

Figure 7 – Probability density functions of TS-FIS prediction errors, for both models, estimated with Parzen windows, for the training set. Plots generated with Parzen windows with $\sigma = 0.01$.

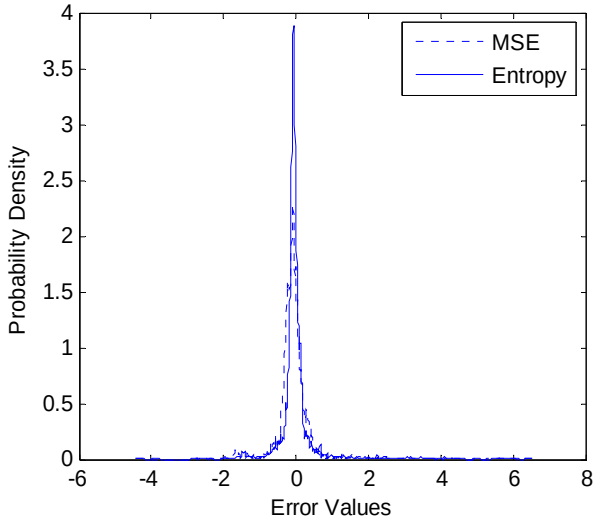

Figure 8 - Probability density functions of TS-FIS prediction errors, for both models, estimated with Parzen windows, for the test set. Plots generated with Parzen windows with $\sigma = 0.01$.

To be able to appreciate the impact of these results on the time domain, we plot on Figure 9 a sequence of values from the test set, including the actual power measure at the SCADA and the predictions produced by the MSE and the ITL models.

This is a crude plot, before a squashing filter converting all values below zero into a prediction of zero output. It is visible in this figure that the curve generated by the Entropy based predictor follows more closely the target (real values) curve that the curve generated by the MSE training objective. This closeness is not uniform over the whole year of data and Figure 10 displays a period of the year where the two models are both very close to the observed curve.

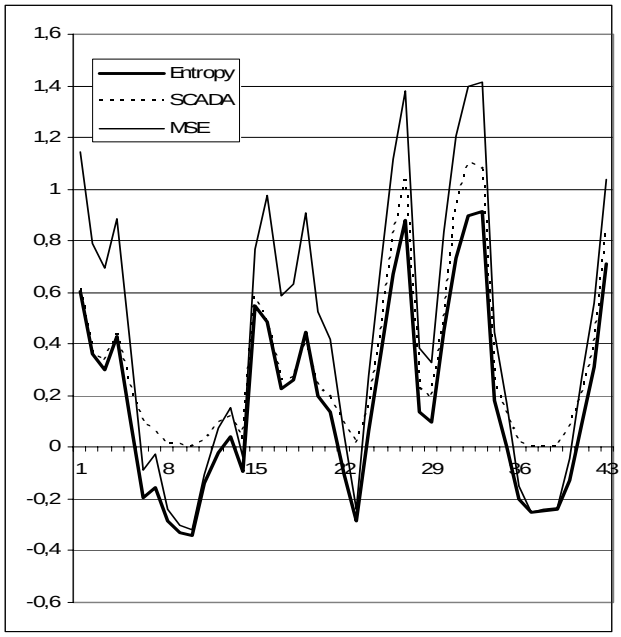

Figure 9 – Comparison, on a subset of the test set (7 weeks), of the performance of the two models. The x axis unit is days, but the plotted values are hour values. The y axis is in p.u. of nominal installed capacity. It is clear that the MSE criterion did not perform as well as the ITL Entropy criterion.

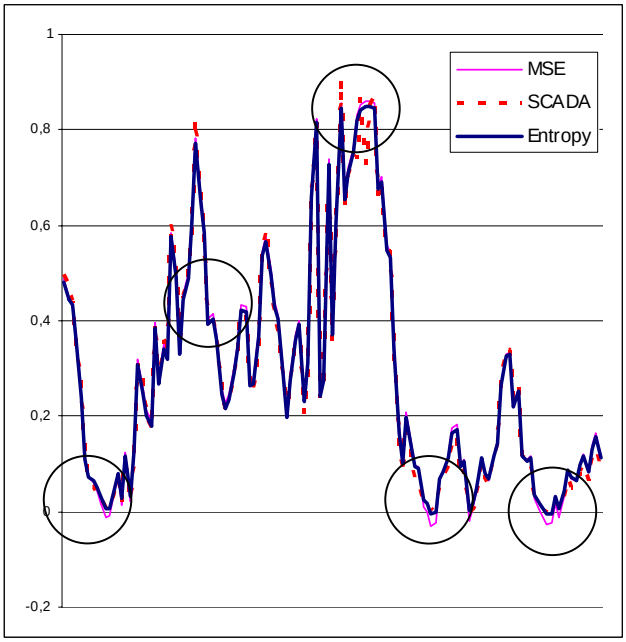

Figure 10 – Comparison, in 9 weeks, of the performance of the two models. The Entropy trained model is still marginally better than the MSE trained predictor, as observed in places marked by circles.

In general, examining the 5000 points we will find some places where the MSE model even produced a smaller error. However, they are outnumbered by the number of cases where the model trained by minimizing Entropy overcomes the model trained by minimizing MSE. This is not surprising because the analysis of the error probability density functions is telling us exactly this: that the Entropy model will generate errors that in general are closer to zero.

IX. CONCLUSIONS

The objective of this paper is to show that the Entropy concept, in the manageable form achieved by the Information Theoretic Learning approach, is a powerful tool with the potential to lead to the development of better prediction models.

Researchers and developers should question the blind application of the Mean Square Error, as a measure of performance of Fuzzy Inference Systems or Artificial Neural Networks, or any other model of reality depending on parameters adjusted with training. The MSE criterion is a quadratic function of the errors and, in fact, it represents the variance of the error distribution. By considering only variance in the optimization of parameters, methods based on the MSE are not sensitive to information contained in moments of higher order in the distribution of errors. In many real problems error distributions are not well behaved nor symmetrical or Gaussian (which are the distributions that contain all information in their first two moments – mean and variance). Therefore, a method that relies on the minimization or error variance is not using to the full extent the information contained in data.

Using Entropy as a performance criterion was not really

manageable until an approach – Information Theoretic Learning – combined it, in the form of Renyi's Entropy, with Parzen windows. Nonetheless, computing this criterion is more expensive than computing the MSE criterion, because the latter only depends on (the square of) errors and the former depends on (the Gaussian of) the differences of errors. For off-line systems, however, this extra effort is worthwhile in the development phase, if it indeed leads to better predictions.

The paper contains also the demonstration of the independence of the three blocks involved in the training of mappers: the mapper itself, the cost criterion and the optimization algorithm. In fact, the fuzzy inference system was not trained by back-propagation but by applying a meta-heuristic EPSO – Evolutionary Particle Swarm Optimization. We could have used a platform such as ANFIS for the application of the MSE criterion, but this would not provide weight calculation for the Entropy criterion. To put all simulations under the same conditions, and not make the comparisons dependent on the method used, we have applied the same algorithm (EPSO) to both models and built Takagi-Sugeno Fuzzy Inference Systems from there. It is also, by the way, the first time EPSO is used for such an application, with success.

The example presented belongs to the intensive research efforts presently done in the area of wind prediction and wind power forecasting. Any technique that helps in extracting more information from data will help, because the problems are very difficult, especially in medium and long term prediction up to 72 hours.

The model presented belongs to the category of wind prediction models that are composed of a series of models, from meso-scale meteorological predictions to the local prediction of wind speeds and finally to the prediction of power output of wind parks. It is not a full prediction model but only a component of a more complex composite wind power prediction system.

Wind prediction is still an exercise with a large uncertainty. In Europe, where wind generation is assuming an important role, models are under development aiming at producing predictions of wind speed up to 72 hours, but average errors are still in the range of 25%. Any small improvement is extremely valuable. Training models with entropy concepts will provide one way to such improvement.

X. Acknowledgment

Work partially developed while visiting the University of Florida, Gainesville, FL, USA, in April 2005. V. Miranda gratefully acknowledges the scientific support of Prof. J. Principe and also the financial support of FCT – Foundation for Science and Technology, Portugal.

XI. References

[1] L. Landberg, "Short term prediction of the power production of wind farms", Journal of Wind Eng. and Ind. Aerodynamics, no.80, pp. 207-220, 1999

[2] B. Bailey, M. C. Brower and J. C. Stack "Short term wind forecasting – development and application of a mesoscale model!, Proceedings of the 1999 European Wind Energy Conference EWEC'99, Nice, France, pag. 1062-1065, March 1999

[3] G. Giebel, L. Landberg and T. S. Nielsen, "The ZEPHYR project: the next generation prediction tool", Proceedings of the 2001 European Wind Energy Conference EWEC'01, pp. 777-781, Copenhagen, Denmark, June 2001

[4] S. Li, D. C. Wunsch E. A. O'Hair, "Using neural networks to estimate wind power turbine generation", IEEE Transactions on Energy Conversion, vol. 13, no.3, pp. 276-282, September 2001

[5] F.A. Castro and J.M.L.M. Palma, "VENTOS (TM): A computer code for simulation of atmospheric flows over complex terrain", Technical Report (available from the authors), http://www.fe.up.pt/ventos, FEUP, Porto, Portugal, 2002

[6] M. Lange, "Analysis of the uncertainty of wind power predictions", PhD Thesis, Carl von Ossietzky University, Oldemburg, Germany, 2003

[7] M. Lange, "On the uncertainty of wind power predictions - Analysis of the forecast accuracy and statistical distribution of errors", Transactions of the ASME, Journal of Solar Energy Engineering, no. 127, v.2, pag. 177-194, May 2005

[8] J. C. Principe and Dongxin Xu, "Introduction to information theoretic learning", *Proc. International Joint Conference on Neural Networks (IJCNN'99)*, Washington DC, USA, 10-16 July 1999, pp. 1783-1787

[9] J. C. Principe and D. Xu "Information-theoretic learning using Renyi's quadratic entropy", in J.-F. Cardoso, C. Jutten, and P. Loubaton, editors, *Proceedings of the First International Workshop on Independent Component Analysis and Signal Separation*, Aussois, France, pages 407-412, 1999.

[10] C.E. Shannon, "A Mathematical Theory of Communications", Bell Systems Technical Journal, vol. 27, pp. 379-423, pp. 623-656, 1948.

[11] A. Renyi, "Some Fundamental Questions of Information Theory", *Selected Papers of Alfred Renyi*, vol 2, pp. 526-552, Akademia Kiado, Budapest, 1976.

[12] E. Parzen, "On the estimation of a probability density function and the mode", *Annals Math. Statistics*, v. 33, 1962, p. 1065

[13] D. Erdogmus and J. C. Principe, "Generalized Information Potential Criterion for Adaptive System Training", *IEEE Transactions on Neural Networks*, vol. 13, no. 5, September 2002, pp. 1035-1044

[14] R. A. Morejon and J. C. Principe, "Advanced search algorithms for information-theoretic learning with kernel-based estimators", *IEEE Transactions on Neural Networks*, vol. 15, no. 4, July 2004, pp. 874-84

[15] D. Erdgomus, K. Hilde and J. C. Principe, "Online Entropy Manipulation: Stochastic Information Gradient", IEEE Signal Processing Letters, vol. 10. no. 8, Aug 2003

[16] D. Erdgomuz and J.C. Principe, "An Error-Entropy Minimization Algorithm for Supervised Training of Nonlinear Adaptive Systems", IEEE Transactions on Signal Processing, vol. 50, no. 10, Jul 2002

[17] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its application to modeling and control", IEEE Transactions on Systems, Man and Cybernetics, v. 15, pp. 116-132, 1985

[18] J.-S. Roger Jang, "ANFIS – Adaptive Network Based Fuzzy Inference Systems", IEEE Transactions on Systems, Man and Cybernetics, vol. 23, no. 3, May 1993

[19] V. Miranda and N. Fonseca, "EPSO – Best-of-Two-Worlds Meta-Heuristic Applied To Power System Problems ", *Proceedings of WCCI 2002 - World Congress on Computational Intelligence - CEC - Conference on Evolutionary Computing*, Honolulu, Hawaii, U.S.A., May, 2002

[20] V. Miranda, N. W. Oo, "New experiments with EPSO – Evolutionary Particle Swarm Optimization", Proceedings of the IEEE Symposium on Swarm Intelligence, Indianapolis (In), USA, May 2006

[21] V. Miranda and N. Fonseca, "EPSO - Evolutionary Particle Swarm Optimization, a New Algorithm with Applications in Power Systems", Proceedings of the IEEE Transmission and Distribution Asia-Pacific Conference 2002, Yokohama, Japan, Oct 2002

[22] V. Miranda and N.W. Oo, "New experiments with EPSO – Evolutionary Particle Swarm Optimization", Proceedings of the IEEE Swarm Intelligence Symposium, Indianapolis, Indiana, May 2006

[23] H. Mori and Y. Komatsu, "A Hybrid Method of Optimal Data Mining And Artificial Neural Network for Voltage Stability Assessment", *Proceedings of IEEE St. Petersburg PowerTech Conference*, Russia, June 2005

[24] N. W. Oo and V. Miranda, "Multi-energy Retail Market Simulation with Intelligent Agents", *Proceedings of IEEE St. Petersburg PowerTech Conference*, Russia, June 2005

[25] D. Erdogmus and J. C. Principe, "Comparison of entropy and mean square error criteria in adaptive system training using higher order statistics", in P. Pajunen andJ. Karhunen, editors, Proceedings of the Second International Workshop on Independent Component Analysis and Blind Signal Separation, Helsinki, Finland, Otamedia, Espoo, Finland, 2000, , pages75-80

XII. BIOGRAPHIES

Vladimiro Miranda (M'90, SM'04, F'05) received his Licenciado, Ph.D. and Agregado degrees from the Faculty of Engineering of the University of Porto, Portugal (FEUP) in 1977, 1982 and 1991, all in Electrical Engineering. In 1981 he joined FEUP and currently holds the position of Professor Catedrático. He is also currently Director of INESC Porto, an advanced research institute in Portugal. He has authored many papers and been responsible for many projects in areas related with the application of Computational Intelligence to Power Systems.

Cristina Cerqueira is graduated in Mathematics Applied to Technology, from the Faculty of Sciences of the University of Porto, Portugal. She joined INESC Porto as a young researcher in 2005, in the Power Systems Unit, working in evolutionary methods of optimization. She contributed to this work in her final graduation year during training at INESC Porto.

Cláudio Monteiro was born in France, on March 14th, 1968. He received his Licenciado and MSc. and Ph.D degrees from FEUP in 1993, 1996 and 2003, in Electrical Engineering and Computers. In 1993 he joined INESC Porto as a researcher in the Power Systems Unit.