Creating Confidence Intervals for Reservoir Computing's Wind Power Forecast

Use of Maximum Likelihood Method and the Distribution-based Method

Breno Menezes. Mêuser Valença
Escola Politécnica de Pernambuco
Universidade de Pernambuco
Recife, Brasil
Emails: (bamm, meuser)@ecomp.poli.br

***Abstract*—The world is increasing the investments in electricity production from renewable sources, such as wind farms, although, the variable power production of wind farms must be balanced by other sources of energy, such as thermal units. As the amount of electric energy generated by the wind represents a higher percentage in the electric grid, it becomes more important to do accurate wind power forecasts and confidence intervals to support the system's operation and reduce its costs. In order to generate the confidence intervals, the forecasting error is often assumed to follow a Gaussian distribution. A wrong assumption can have a huge impact on the confidence intervals. This work proposes an evaluation of the forecasting error distribution generated by a Reservoir Computing network forecast in different timescales and the confidence intervals generated using the maximum likelyhood method and the distribution based method.**

Keywords-Reservoir Computing; Confidence Intervals; Maximum Likelihood Method; Distribution Fit.

I. INTRODUCTION

Following the ideas of sustainable growth, the World is increasing the use of wind to generate electrical energy. As an alternative to fossil fuels, it is plentiful, renewable and widely distributed around the globe [1]. The problem involving this source of energy is that the wind might be inconsistent as it is strongly influenced by the weather conditions and other sources of energy are needed to cover the deficit from wind farms.

As the participation of wind farms in the electrical grid is raising, the challenge to maintain the balance between power generation and load is even harder. Another sources of energy must be kept in order to compensate eventual changes in the wind power output. In this scenario, it is very important to have accurate wind power forecasts. As the forecast improves its accuracy, the need of other sources of energy, such as thermal units, are minimized, representing a cost reduction to the system. The cost of wind power forecast errors in a single plant is around €15,000 - 18,000 per MW of installed capacity [2], which is the production capacity of a plant.

In order to improve the wind power forecast, which is mostly represented as a single value, confidence intervals can be built. They represent a range of estimated values for a certain probability. These intervals increase the amount of information of the prediction, making it easier to operate the system. There can be different intervals for distinct timescales. This kind of estimation is important because the system must be able to anticipate the needed load by increasing the generation from other sources, such as hydroelectric power, like Brazil, or slow starting thermal units. And this is why it is really important to have forecasts done with results for every hour to one day ahead.

The generation of confidence intervals is highly correlated to the forecast error distribution. It is very common to assume that the errors can be represented by a Gaussian distribution in every timescale. This assumption can create errors and the system may overestimate or underestimate wind power generation, and this represents extra costs to the system operation [2].

Some studies have concluded that these errors are better represented using a Cauchy or Weibull distribution when compared to a Gaussian or Beta distribution [3]. One of the objectives of this work is to study the error distribution of a wind power forecast generated by a Reservoir Computing network using data from a real wind farm in Brazil. This work will evaluate which distribution represents better the error in different timescales doing a distribution fit, in order to improve the forecast (Figure 1).

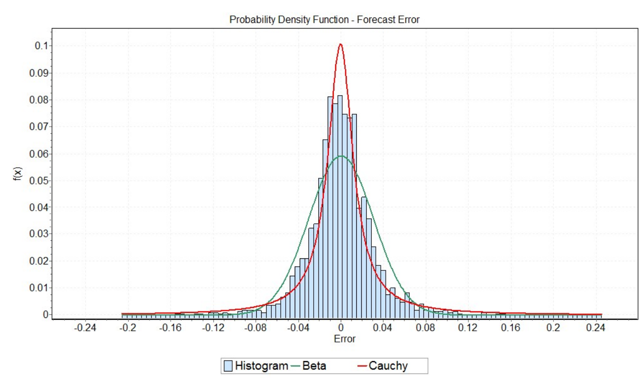

Figure 1. Normal vs. Cauchy Distribution.

After finding the distribution that fits better the forecast errors, it will be possible to compare the confidence intervals generated. The method used to generate the interval is proposed by Nandeshwar [4], the maximum likelihood method. The method assumes that a prediction has two uncertainty sources. The first one is related to the prediction error and the second one is related to the error generated by the network

 ISBN: 978-1-61208-340-7

when trying to predict its own error. The maximum likelihood method uses these two values to generate an upper and a lower limit to represent the confidence interval. The final objective is to compare and evaluate the confidence intervals generated for distinct timescales using values from the distribution that fit better with forecast values.

This paper is organized as follows. Section II contains the basic explanation about *Reservoir Computing* and its training. All the information about the distribution fit problem is explained in Section III. Details about confidence intervals and the methods to generated them used on this paper are reported in Section IV. Section V explains the methodology used, giving details about the database used, how the network was trained and how the distribution fit was done. The results are shown on Section VI and the conclusions in Section VII.

II. Reservoir Computing

Artificial Neural Networks are mathematical models of computer intelligence based on the behavior of the human brain and its basic unit, the neuron. Inspired by the biological neuron, the artificial neuron was modeled. It is capable of processing entering signals (inputs) and generating one or more outputs, representing a synapse [5].

A Multi Layer Perceptron network (MLP) determines new rules for neurons' organization. Unlike the previous networks, it has a hidden layer. This new layer is responsible for the non linearity of MLP, making it possible to solve non-linearly separable problems, like the most of real problems.

The Reservoir Computing is presented as a network of several layers including the principle of recurrences in its links. Thus, the network is no longer feed-forward and the signal propagates through the network in several directions.

One of the ideas that gave rise to the Reservoir Computing network was the Echo State Network [6]. It is characteristic of it that a neuron in the middle layer can be interconnected, including with itself. This topology creates cycles within the network, and the signal propagated "echoes" by these recurrences. The network creates a kind of signals memory. The signals lose strength with time as they propagate through cycles (Figure 2).

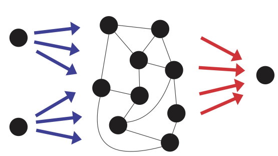

Figure 2. Reservoir Computing Network.

The topology of the network does not allow the use of simple training methods as used in MLP, the backpropagation method. The difference is caused by the recurrences. The method used in this work uses the matrix inversion (JAMA) to adjust the weights connected to the output layer. The other weights are determined once and remain the same during the whole training.

With these characteristics, Reservoir Computing is proposed to be a powerful tool to solve dynamic problems related with time processing and continuous values just like the wind power prediction [7], and this is why it was chosen. Because of its recent implementation, some of the Reservoir Computing's properties are not theoretically based and they are set empirically.

III. Distribution Fit

As the Reservoir Computing returns a set of predicted values using the input data, the error can be calculated subtracting it from the real values for each output. Some information of theses errors need to be extracted in order to generate the confidence intervals. One of them is the error distribution, which is often assumed to be Gaussian/Normal, and it is not true for all the cases. This kind of mistake can have a huge impact on the confidence interval accuracy [2].

Probability distribution fitting, or just distribution fitting, is a way to fit a probability distribution function to a data set by observing data characteristics to adjust the distribution's parameters. Doing this it is possible to predict the probability of occurrence of a certain phenomenon. There are many distributions of which some can represent a data set better than others, depending on the data characteristics. The distribution giving a close fit is supposed to lead to good predictions.

The distributions evaluated in this work are normal, beta, cauchy and weibull distribution. Each of these will be fit to the set of errors generated by the network and the results will be compared. The Kolmogorov-Smirnov test will be used to decide which distribution fits better to the set of errors. As the model have predictions for every half an hour for 24 hours, there can be a variation of which distribution fits better depending on the timescale. The prediction may present different results for a short timescale (+00:30 or +01:00) when compared to a long range timescale (+24:00) for example.

IV. Confidence Intervals

In a prediction model, the output value does not represent much information to the user. To use it in a proper way there is a need to know how much reliable the model is, or how certain it is about the outputs. Because even a unstable model may present random good results for a period of time.

The confidence intervals have as its objective measure the reliability and also aggregate information to the results presented. Instead of estimating one single value, a interval of probable estimates is given. The range of this interval will determine the quality of the results presented for a specific problem. If the interval is too wide, it means that the reliability is low and the model shows unstable results. If the interval is thin, it means that the results are constant and the model is reliable.

With higher probabilities wider will be the intervals. But if it is too wide, it means that the results are poor on information about the problem. In a ideal situation, the model will generate a thin interval for a high value of probability of containing the occurred value. The value of probability is chosen by the needs of the user.

In other models based on neural networks, confidence intervals are implemented using traditional linear statistical models [8].

In neural networks prevision based models, the confidence intervals have generally been set grounded by traditional linear statistic models [8]. However, as neural networks are non-linear models those linear methods are not always appropriate so that new proposals are being tested, such as the maximum likelihood method.

A. *Maximum Likelihood Method*

To create a confidence interval, it is necessary to set an upper and a lower limit from the chosen probability and the results obtained in network training.

In this work, we applied the maximum likelihood method [4] proposed by Nandeshwar. The method assumes that there are two sources of uncertainty in a interval prediction model:

- The first is the noise variance (σ_v^2), calculated from the variance calculation standard equation (2) applied to absolute errors (1).
- The second is the uncertainty's variance (σ_w^2), calculated from a separate network variance (2) errors that aims errors generated by the first network training. This second variance is related to the network's ability to predict the error itself.

$$Error = |occurred\_value - calculated\_value| \tag{1}$$

$$\sigma = \frac{1}{n-1}\sum_{i=1}^{n}(Error_i - \widehat{Error})^2 \tag{2}$$

where:
$Error_i$ = Errors regarding the i entry.
$\widehat{Error}$ = Errors average.

The total variance is determined by the sum of the two previously calculated variances (3).

$$\sigma_{total}^2 = \sigma_v^2 + \sigma_w^2 \tag{3}$$

The calculation of the confidence interval is done using (4). Given a predicted value f(x):

$$f(x) - t * \sigma_{total}^2 < f(x) < f(x) + t * \sigma_{total}^2 \tag{4}$$

where t is the extracted value of the t Student's table. In this work the largest possible degree of freedom has been chosen combined with an alpha value of 5%.

The algorithm on Figure 3 shows the steps to generate the confidence interval using the maximum likelihood method.

1. Divide the database into training data and test data;
2. Normalize data;
3. Train the network;
4. Calculate the error variance σ_v^2 (2);
5. De-normalize data;
6. Calculate errors (calculated - occurred);
7. Normalize data again;
8. Train a new network with the same inputs, aiming errors calculated in step 6;
9. Calculate this network's error variance, σ_w^2 (2);
10. De-normalize data;
11. Calculate σ_t^2 (3);
12. Select the appropriate confidence level and find its value in the t Student's table;
13. Calculate the range for the values obtained with the test data (4).

Figure 3. Maximum Likelihood Method.

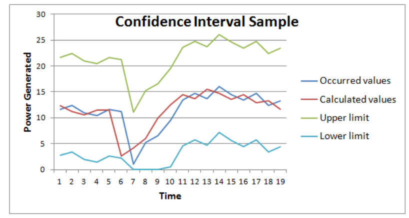

Figure 4. Confidence Interval sample.

The confidence interval validation is made using the occurred and the calculated values. When applying the interval, the occurred values must be within the range of the interval with the same probability used to choose the value of t in the t Student's table (Figure 4).

V. METHODOLOGY

A. *The Database*

In order to achieve the objectives of this work, a real database was used. The data come from a real wind farm in Brazil containing measurements for every 30 minutes including wind speed, direction and generated power. Each value represent a mean calculated from every windmill in the farm. They also include the date it was measured, the standard deviation, minimum and maximum value for each type of data. The installed capacity of this wind farm is 54.61MW.

In total, there are 11712 measurements. Therefore, a few values are missing due to problems during the measurement. This instants are not using during the network training. All values chosen to be the inputs on the network's training are values of mean generated power. The whole data is spited in 3: training, cross validation and testing set.

Before using the data for training, it must be normalized (step 2 on the algorithm show on Figure 3). This is done using a linear transformation 5. The data will be represented on a interval between 0.25 and 0.75.

$$x = \frac{(x - x_{min}) * (b - a)}{x_{max} - x_{min}} + a \tag{5}$$

where:
x_{max} = Maximum value found on the real data.
x_{min} = Minimum value found on the real data.
a = Minimum value for the normalized data.
b = Maximum value for the normalized data.

Doing this the network will deal with values that are alike attributing the same importance to a low and a high value. The de-normalization (step 5 on the algorithm show on Figure 3) is done using this same equation in a reverse way.

B. *Reservoir Computing*

The Reservoir's structure is an important factor that could determine a good or bad performance for the wind power forecast task. As being a result of recent studies, some network properties are determined empirically.

In this work, a Reservoir Computing network with three layers was created. The first one represent the inputs. To set the number of inputs, the linear correlation was evaluated and it was decided to use 7 past values to be the inputs. Too much past values may confuse the network and too few values may not give enough information to the network. The hidden

 ISBN: 978-1-61208-340-7

layer was set up with 20 neurons determined empirically. Both of these two layers have the sigmoid logistic as the activation function. The output layer has 48 neurons, each one representing one of the 48 instants of half an hour a head that are needed to be predicted. The number of outputs has been chosen according to other existing methods that are already applied on ONS (Brazilian electric system operator).

On the hidden layer, there is a 20% probability of existing a connection between two neurons where it could be positive or negative by the same chance. This probability value determines the amount of recurrences in the network.

Based on previous work this Reservoir Computing network has a little improvement from the basic version. It has connections between the input layer and the output layer. This feature leads to better results.

The training uses 75% of the database, where 50% is used for initial training and 25% is used for cross-validation. The rest is used for tests where values that never have been shown to the network are used to evaluate its results.

The training can be stopped if a maximum number of cycles have occurred or if the network is not improving itself anymore. For this evaluation, the mean squared error is calculated using (6). If the error stops decreasing it means that the network has been stable and need no further training.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(x_{calc_i} - x_{obs_i})^2 \qquad (6)$$

The last part of training is the test phase. The predicted values generated on the test phase are used to evaluate the results of the network and also to create the confidence intervals. The error for each timescale is calculated by subtracting the real value minus the predicted value, doing this we get a set of errors for each one of the outputs. As the Reservoir presents predictions for distinct timescales, the confidence intervals may present different behaviors. For each one, the distribution fit is applied in order to find a probability distribution to represent the errors in a certain timescale and generate a better confidence intervals.

C. Distribution Fit

The distribution fit can be done by adjusting the parameters of a certain probability distribution using the characteristics of a set of values. In this work, the normal, beta, cauchy and weibull distribution are adjusted to the sets of errors obtained on the Reservoir's training. To evaluate which one is more similar to the error set, the Kolmogorov-Smirnov test is used. This test is used to compare a sample with a reference probability distribution.

Using this method, it is possible to rank the four distributions for errors in each timescale. As the errors present distinct behaviors for different timescales, it is possible the rank have variations for each scale and this will make it possible to choose the best distribution for each case.

VI. Results

A. Maximum Likelihood Method

In order to evaluate the behavior of the maximum likelihood method applied over the prediction of generated power done by a Reservoir Computing network, the hit rate of the confidence interval must be observed. This means that the number of real values inside the boundaries created by the confidence intervals must be coherent to the confidence level chosen, in this case 95%.

TABLE I. Maximum Likelihood results table.

Maximum likelihood method results				
	Sigma V	Sigma W	Sigma T	Hit Rate
+00:30	4.311594	4.302589	6.09115	96.71%
+12:00	13.61194	13.5954	19.2358	99.63%
+24:00	11.8412	11.8304	16.74022	98.56%

The results can be observed on Table I. As the prediction has different properties for distinct time scales the results are shown in 3 moments: Half hour ahead, 12 hours ahead and 24 hours ahead.

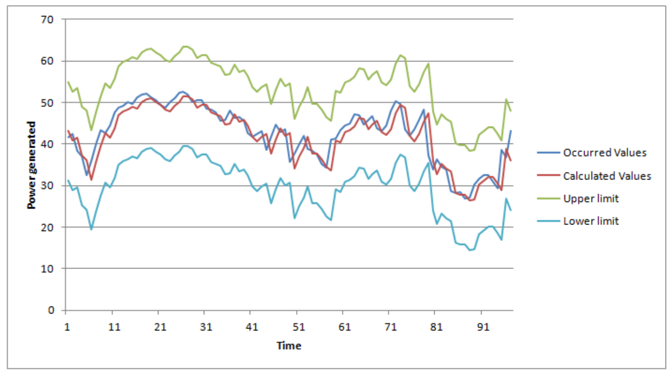

Figure 5. Maximum likelihood interval for +00:30.

The results show that the predicted values are really close to the real values (Figure 5) and the prediction error present low variance for a short range of time (+00:30). This statement can be observed on the corresponding line on Table I.

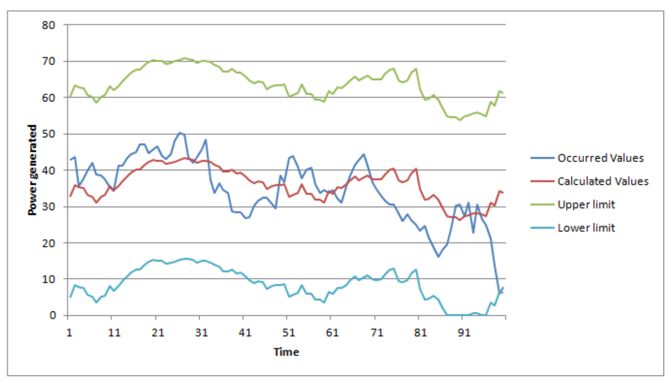

Figure 6. Maximum likelihood interval for +24:00.

For instants that are far ahead, 12 or 24 hours ahead, and the correlation with past values is lowered the Reservoir does not show constant results and the error variation is increased (Table I). For this reason, the confidence intervals generated are wider (Figure 6) and the hit rate is increased because of the interval's range. This means that the Reservoir has poor results on these timescales, which can be justified by the low correlation caused by the randomness of the wind in a larger

TABLE II. Distribution Fit +00:30 results table.

	Kolmogorov-Smirnov	
Distribution	Statistic	Rank
Cauchy	0.05301	1
Normal	0.07168	2
Beta	0.0719	3
Weibull	0.12251	4

TABLE III. Distribution Fit +12:00 results table.

	Kolmogorov-Smirnov	
Distribution	Statistic	Rank
Beta	0.03576	1
Weibull	0.04459	2
Normal	0.04569	3
Cauchy	0.10767	4

timescale, compromising the prediction and the confidence interval's quality.

B. Distribution Based Method

Before generating the confidence intervals using the distribution based method, the distribution fit must be evaluated using the Kolmogorov-Smirnov test. It may be observed that for each timescale there was a distribution that fits better with the errors set (Tables II, III, IV), and this one will be used as a reference to generate the confidence interval.

For half an hour ahead (+00:30), the probability distribution that fit better with the forecast error distribution was cauchy (Figure 7). The K-S test pointed it as being the most similar between to the error distribution. The normal and beta distribution had similar results and weibull showed the poorest result for this time scale (Table II).

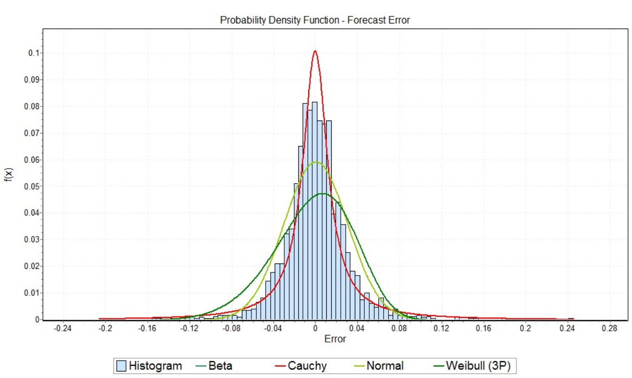

Figure 7. Distribution Fit +00:30.

TABLE IV. Distribution Fit +24:00 results table.

	Kolmogorov-Smirnov	
Distribution	Statistic	Rank
Weibull	0.01944	1
Beta	0.02041	2
Normal	0.02492	3
Cauchy	0.09133	4

For Twelve hours ahead (+12:00), cauchy did not present the same results as for half an hour ahead, it was actually the worst in the Kolmogorov-Smirnov test (Table III). For this timescale the winner distribution was beta followed by weibull and normal in this order (Figure 8).

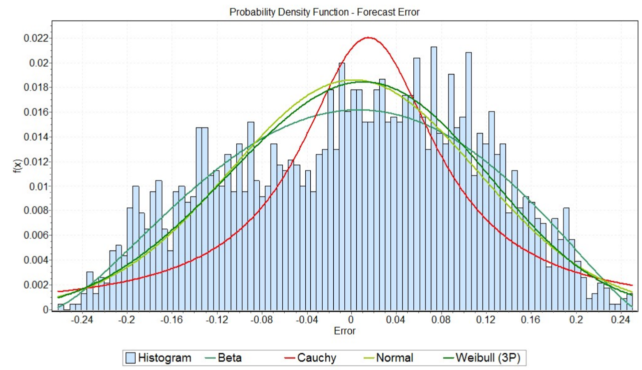

Figure 8. Distribution Fit +12:00.

For twenty four hours ahead, presenting a large variation between the predicted values and the real values, the distribution that fit better with the error distribution was weibull (Table IV), followed by beta, normal and cauchy (Figure 9).

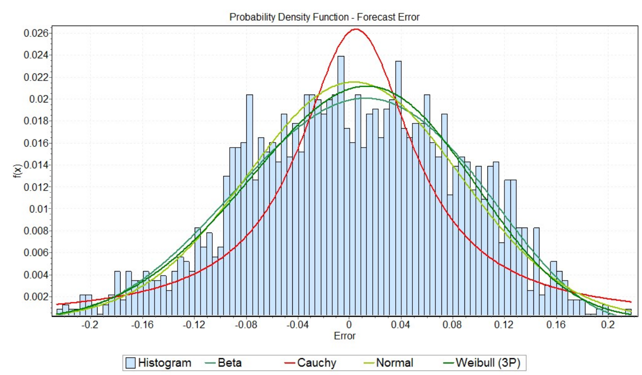

Figure 9. Distribution Fit +24:00.

Having the results previously mentioned, it is possible to generate the confidence intervals based on the right probability distribution. As seen, the error may be represented better by a certain distribution in a timescale, and this one will be chosen to create the intervals for that instant.

Generated for a half an hour ahead timescale, the first confidence interval will use the cauchy distribution (Table II). As it is fit to the error distribution, the values are extracted from it according to the confidence level chosen, 95%. Being the distribution asymmetric, two values must be extracted, one for the positive side and one for the negative side. These values are -0.164 and 0.184. To establish the upper and lower limit of the confidence interval these values must be multiplied with the installed capacity of the wind farm and then added the calculated value (Reservoir's output).

Part of the interval for +00:30 may observed on Figure 10. It was obtained a hit rate of 98.2668% (real values inside the confidence interval's range).

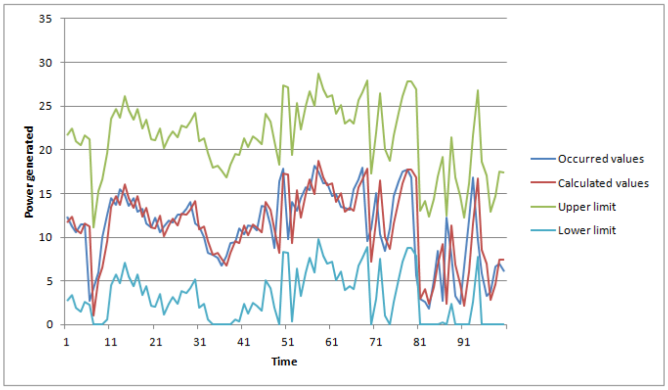

Figure 10. +00:30 Confidence interval sample.

Using beta distribution, the confidence interval generated for a +12:00 timescale may be observed on Figure 11. The values extracted from the distribution are -1.66 and 1.64, also following the 95% confidence level. The hit rate was 88.99%.

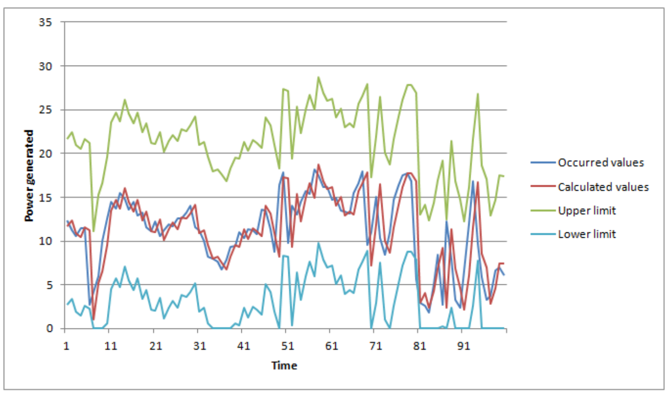

Figure 11. +12:00 Confidence interval sample.

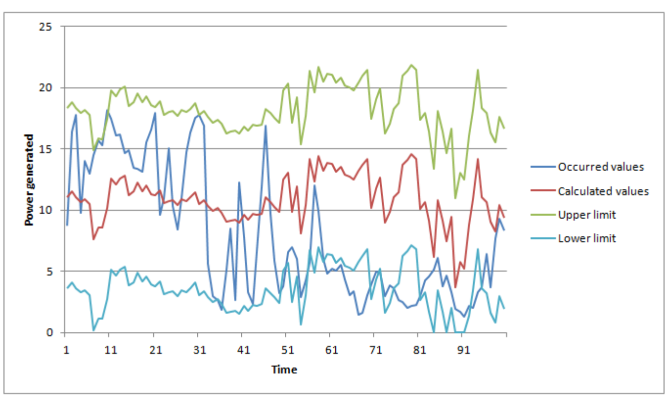

Figure 12. +24:00 Confidence interval sample.

For twenty four hours ahead (+24:00) and using the weibull distribution, a sample of the confidence interval for this timescale may be observed on Figure 12. The values extracted from this distribution are -1.36 and 1.33 and the hit rate is 90.1646%.

VII. Conclusion and Future Work

Analyzing the presented results, it is possible to take a few conclusions about the generation of confidence intervals applied on the wind power forecast done with Reservoir Computing. Also, it is possible to compare the two methods used in this work.

The maximum likelihood method showed coherent results for a short range timescale (+00:30). The confidence interval has a hit rate close to what it was proposed (95%) and it is not too wide. For the cases where the Reservoir Computing did not generate stable results and the error variation was high (+12:00 and +24:00), the maximum likelihood method did not show good results. As the variation is high, the intervals show a wide range for the same 95% confidence level.

The distribution based method presented good results for all three timescales. In all of them the confidence interval's width is quite similar and the hit rate is close to what it was proposed, the 95% of confidence level. This proves that a distribution analysis is worth done in order to improve the generation of the confidence intervals.

Comparing the result of these two techniques, it is possible to conclude that they show similar results for sets that present a distribution with low variation. In the other case, where there is some large variation, the distribution fit helped the creation of better confidence intervals with the same confidence level but with a shorter range.

In the future works, these techniques will be applied to other set of data that have different characteristics in order to compare the results. Also, another model must be tested in the prediction to check if this result are only related to the Reservoir Computing's outputs. Other techniques to generate confidence intervals must be evaluated.

References

[1] A. L. Sá, "Wind energy: principles and aplications." [Online] Avaliable from: http://www.cresesb.cepel.br/ , acessed on August 09th, 2012.

[2] B. Hodge and M. Milligan, "Wind power forecasting error distribution over multiple timescales," *To be presented at the Power & Energy Society General Meeting*, 2011.

[3] B. Hodge, A. Florita, K. Orwig, D. Lwe, and M. Milligan, "A comparison of wind power and load forecasting error distributions," *Presented at the 2012 World Renewable Energy Forum*, 2012.

[4] A. R. Nandeshwar, "Models for calculating confidence intervals for neural networks," Master's thesis, West Virginia University, 2006.

[5] M. J. S. Valenca, *Fundamentos das Redes Neurais*. Livro Rapido, 2011.

[6] M. Lukosevicius and H. Jaeger, "Reservoir computing approaches to recurrent neural network trainning," *Computer Science Review*, vol. 3, August 2009, pp. 127–149.

[7] D. Verstraeten, *Reservoir Computing: computation with dynamical systems*. PhD thesis, Ghent University, 2009.

[8] D. S. Moore, *The Basic Practices of Statistics*. W.H. Freeman and Company, 3rd ed., 2005.