uc3m | Universidad Carlos III de Madrid

e-Archivo

This document is published at:

The most commonly used ways of representing uncertainty are quantiles and prediction intervals (PIs). The estimation of quantiles has been widely addressed in the literature, especially through methods that estimate each desired quantile independently. For example, quantile regression (*QR*) [7] constructs linear models by minimizing quantile loss (or pinball score) for a particular quantile-probability. Nonlinear models can also be constructed using this cost function, for example quantile regression neural networks (*QRNN*) [8] or tree-based ensembles such as Gradient Tree Boosting (*GB*) [9,10]. There are also techniques which build a single model from which all quantiles can be extracted. For instance, quantile regression forests (*QRF*) [11] is another non-linear ensemble technique and analog ensembles (AnEn) [12–14] generate probability distributions using a set of past

measurements that correspond to the most similar past numerical weather prediction (NWP) forecasts to the current situation.

Another way of representing forecast uncertainty is by means of PIs, where two bounds, lower and upper, contain the actual value with a given probability (called prediction interval probability coverage (PICP)) [15]. For a given PICP, the narrower is the PI, the less uncertain is the forecast. There are in the literature many classic methods for computing PIs [16], but, more recently, an evolutionary approach, known as LUBE (lower upper bound estimation), has shown better performance in several domains [16], including renewable energy forecasting [17,18]. The approach uses a two-output artificial neural networks, for the lower and upper bounds of the interval, respectively. For point forecasting, the actual output to the networks is available, and, thus, supervised training techniques, such as backpropagation can be used. However, this is not the case for LUBE, because the two outputs of the network (i.e., the lower and upper bounds) are not directly available. For this reason, the network parameters are usually optimized using evolutionary computation techniques such as simulated annealing (SA) [19] or particle swarm optimization (PSO) [20]. Optimizing PIs is clearly a multi-objective problem, because obtaining high-coverage but narrow intervals are conflicting goals. However, most LUBE approaches translate the multi-objective problem into a single-objective one, by aggregating the two goals, but this requires some reasonable weighting between the objectives to be decided in advance. LUBE can also be addressed as an actual multi-objective approach that does not require goal weighting. This was the approach proposed by the authors in a previous work [21], where a multi-objective particle swarm optimization evolutionary algorithm (MOPSO) showed very good performance in a solar energy production forecasting problem on Oklahoma solar sites [22], where only meteorological forecasts were used as input. An important advantage of using a multi-objective algorithm is that in a single run the method is able to return a whole set of solutions (Pareto front), which represent the best trade-offs between coverage and width, out of which the user can select a solution for some particular coverage.

Most works that deal with energy prediction use meteorological forecast variables as inputs for the prediction models. However, some articles indicate that prediction error can be improved by using actual measurements available at prediction time, in addition to meteorological forecasts [23,24]. For instance, a solar plant that, at time t_0, needs to issue a power output forecast for the next hour ($t_0 + 1h$) can apply a model that uses (as inputs) meteorological forecasts for $t_0 + 1h$ as well as the current power output (measured at t_0). Martín-Vázquez et al. [25] showed that, in fact, the use of measured output wind electrical power was helpful to improve point forecasts for short horizons in wind energy forecasting. However, not much research has been done to study the influence of using measurements for constructing PIs, although preliminary results point in that direction [26].

The purpose of this article is to study whether the quality of PIs for PV power forecast can be improved by using measured solar output power in addition to meteorological forecast variables. Given that it could be expected that measurements at t_0 may influence short horizons more strongly than longer ones, in this work, several forecast horizons are considered. In principle, using measured information at prediction time should reduce the uncertainty of forecasts (and, therefore, PIs should be narrower), but this advantage is expected to decrease for long prediction horizons. Thus, the main contributions of this article are to study both the degree of improvement of PIs when using measured information and the dependency of that improvement on the prediction horizon. We used the LUBE approach optimized with *MOPSO* [21]. The *MOPSO* approach has already been used for estimating PIs for the aggregated daily energy production forecast for next day [21]. However, in the context of solar forecasting, it is also important to provide uncertainty forecasts for intra-day horizons, which is what we do in the current work. We also used for comparison purposes two benchmark methods recommended in recent literature for probabilistic solar forecasting [4]. The first is *QR* [7], a commonly used linear method for obtaining quantiles in solar forecasting [27]. The second one is *QRF* [11], a non-linear method successfully used in recent works on solar energy uncertainty estimation [28,29].

All methods were tested using as inputs only the meteorological variables and using the meteorological variables together with measured solar power at t_0.

The study was done using data from the Global Energy Forecasting Competition 2014 (GEFCom2014) concerning the probabilistic solar power forecasting problem [30]. To advance beyond the preliminary results [26], in this study, extensive experimentation was carried out. Data from three different solar plants in Australia (Stations 1–3) were used and the PIs were estimated for five intra-day forecasting horizons (1–5 h, UTC time). Given that *MOPSO* and *QRF* are stochastic techniques, they were run 90 times and statistical tests for comparison were done by means of the Wilcoxon test. The extensiveness of the experiments and the statistical test allowed extracting reliable conclusions.

The rest of the article is organized as follows. Section 2 describes the dataset used for experiments. Section 3 summarizes the evolutionary multi-objective approach for interval optimization. Section 4 describes the two baseline methods for comparison. Section 5 describes the experimental setup and the results, including statistical significance tests. Conclusions are finally drawn in Section 6.

2. Data Description

The data used in this work were obtained from Task 15 in the probabilistic solar power forecasting problem [30] one of the four tracks from the Global Energy Forecasting Competition 2014 (GEFCom2014). These data include measured solar power generation and meteorological forecasts for three solar stations (Stations 1–3) in Australia whose exact location was been revealed. Measured solar power generation, expressed in proportions between 0 and 1 relative to the nominal power of the plant, was provided hourly, from 1 April 2012 01:00 to 1 July 2014 00:00 UTC. The meteorological forecasts were obtained from the European Centre for Medium-range Weather Forecasts (ECMWF) and include 12 weather variables. Forecasts for those weather variables are issued by ECMWF once per day, at midnight, and they are forecasted for the each hour for the next 24 h. Thus, if t_0 is the time at which forecasts are issued (00:00 UTC), each meteorological variable gets a forecast for $t_0 + 1$, $t_0 + 2$, $\dots$, $t_0 + 24$.

The 12 weather variables are: total column liquid water, i.e., the vertical integral of cloud liquid water content ($kg \cdot m^{-2}$); total column ice water, i.e., the vertical integral of cloud ice water content ($kg \cdot m^{-2}$); surface pressure (Pa); relative humidity with respect to saturation at 1000 mbar; total cloud cover (0–1); 10-m U wind component ($m \cdot s^{-1}$); 10-m V wind component ($m \cdot s^{-1}$); 2-m temperature (K); surface solar radiation downward ($J \cdot m^{-2}$); surface thermal radiation downward ($J \cdot m^{-2}$); top net solar radiation, i.e., net solar radiation at the top of the atmosphere ($J \cdot m^{-2}$); and total precipitation (convective precipitation + stratiform precipitation) (m).

Figure 1 displays a box plot of the normalized measured power generation for each station corresponding to the whole period (2012-04-01 01:00 to 2014-07-01 00:00) and broken down by UTC hour. It is a standard box plot where the lower and upper hinges correspond to the first and third quartiles and the center black horizontal bar is the median. Whiskers are used to denote data in the 1.5*inter-quartile range and black points are used for outliers. It is important to remark that the hour on the x-axis corresponds to EMCWF time. Given that GEFCom2014 did not disclose the actual location of the three solar stations, it is not possible to know with certainty the local time for each of the stations. It is estimated that noon must correspond to the maximum of solar radiation (Time 1–2 for Station 1; Time 3 for Station 2; and Time 2 for Station 3). Zero is the hour at which ECMWF forecasts are issued. Based on the box plot in Figure 1, and given that measured power should be expected to have influence mostly for short time horizons, it was decided to study Hours 1–5. This also excludes hours too close to the night period.
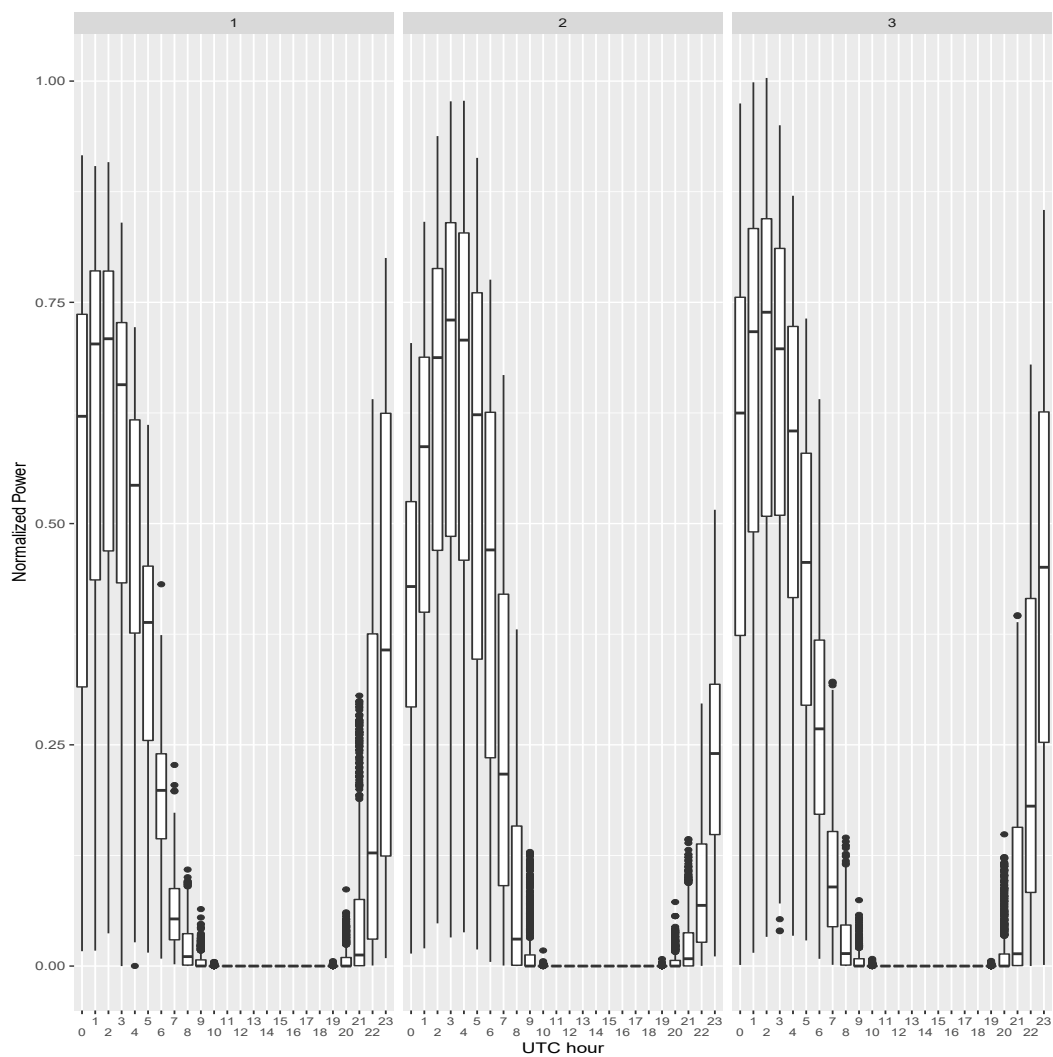
Figure 1. Box plot of solar normalized power, broken down by hour, for: Station 1 (**left**); Station 2 (**middle**); and Station 3 (**right**). The x-axis displays UTC hour.

Given that, for each horizon, there is a single forecast everyday, the available data are made of 820 instances, corresponding to 820 days. To ensure that both train and test sets are representative, every 30 days, 20 consecutive days were chosen for training and the remaining 10 for testing. Therefore, there were 550 days for training and 270 days for testing. To perform hyper-parameter tuning tasks, such as choosing the best neural network architecture, the best number of optimization iterations or the best configuration for *QFR*, it is necessary to split the available training data into another two sets: the training set and the validation set (also known as the development (dev) set). In this work, for every 20-day block of the original training partition, the first 14 days were finally used for training, and the remaining six for validation.

3. Multi-Objective Optimization for Prediction Intervals

The following section describes the multi-objective particle swarm optimization algorithm (*MOPSO*) applied to PI, originally reported in [21]. This application was inspired by LUBE [16], which used an artificial neural network (ANN) with a single hidden layer to make an estimation of the lower and upper bounds of the PIs. The ANN was optimized using multi-objective evolution techniques.

In this study, the inputs to the ANN can be meteorological forecast variables and measured solar output power. The outputs of the network are the boundaries (lower and upper) forecasted with the network for the given inputs. For each input in the dataset, an observation of power is found. However, the boundaries are never included, therefore a backpropagation optimization is not suitable for this network. As the desired outputs are unknown, this is not a standard supervised regression, and an alternative optimization technique is required.

Therefore, an evolutionary optimization algorithm was chosen to optimize the weights of the network with two primary goals: PI coverage and interval width. Coverage is measured using the prediction interval coverage probability (*PICP*), which computes the frequency of observations laying within the interval. The PICP can be maximized by also maximizing the width of the PI, producing trivial solutions. To avoid this, a second goal was set to minimize the width of the PI. The following paragraphs formalize the formulation of these goals.

Let $M = \{(X_i, t_i)_{i=0\cdots N}\}$ be a set of observations, where X_i is a vector with the input variables and t_i is the observed output variable. Let $PI_i = [Low_i, Upp_i)$ be the prediction interval for observation X_i ($[Low_i, Upp_i)$ are the outputs of the ANN). The *PICP* goal can be calculated as described in Equation (1) and the average interval width (*AIW*) in Equation (2).

$$PICP = \frac{1}{N} \sum_{i=0}^{N} \chi_{PI_i}(X_i) \tag{1}$$

$$AIW = \frac{1}{N} \sum_{i=0}^{N} (Upp_i - Low_i) \tag{2}$$

where N is the number of samples, $\chi_{PI_i}(X_i)$ is the indicator function for interval PI_i (it is 1 if $t_i \in PI_i = [Low_i, Upp_i)$ and 0 otherwise), and Upp_i and Low_i represent the PI boundaries.

Optimizing *PICP* requires sacrificing a higher *AIW*, while a narrow *AIW* will lose some coverage in return. The goals are opposed and a multi-objective approach was applied [21]. Every particle in the *MOPSO* represents a single solution to the network, i.e., a different configuration of weights. The goals of this algorithm are: $1 - PICP$ (Equation (1)) and *AIW* (Equation (2)). This network receives the meteorological data (from the given dataset in Section 2) as inputs and measured solar power at t_0.

The *MOPSO* algorithm produces a set of multiple ANNs laid in a Pareto front of non-dominated solutions. Typically, PIs with a desired coverage, called PIs' nominal coverage (or *PINC*), are needed by the user. Sometimes, it is more appropriate to refer to $1 - PINC$, which is named as *alpha* from now on. In this approach, the solution in the Pareto front with *alpha* closest to the one requested by the user is selected [21].

4. Benchmark Methods

To have a baseline to compare *MOPSO* results, linear *QR* [7] and non-linear *QRF* [11] were used.

4.1. Quantile Regression

The *QR* [7] algorithm is able to forecast quantiles by using linear models. In comparison, the least squares method makes a prediction of the target conditional average, while *QR* predicts the median or other quantiles. This is done by minimizing the quantile loss instead of the quadratic loss. Quantiles can be used for obtaining PIs. Let q_1 and q_2 be the $\frac{1-PINC}{2}$ and $\frac{1+PINC}{2}$ quantiles, respectively. There are two quantiles q_1 and q_2 for the left and right $\frac{1+PINC}{2}$ probability tails of the distribution, respectively. The coverage of the PI $[q_1, q_2]$ is equal to *PINC*. To summarize, *QR* builds a couple of linear models to estimate the q_1 and q_2 quantiles, forming the $[q_1, q_2]$ PI in turn.

4.2. Quantile Regression Forest

QRF [11] is an adaption of the random forest (*RF*) algorithm [31] for estimating quantiles and follows a different strategy to linear quantile regression. RF is an ensemble learning algorithm whose

individual models are regression trees. Each of the trees is obtained from different subsets of the training data by means of bootstrap sampling. To obtain an output for an instance, it is dropped down the tree until it reaches a leaf. The average of the response value of the training instances in the leaf is returned. RF can become *QRF* by first adapting the regression trees by changing the information that is stored in the leaf nodes. In the case of *QRF*, leaf nodes store not only the average response, but all the response values of the training instances that reached the leaf. When an instance is dropped down each of the trees in the *QRF* ensemble, every regression tree returns a set of response values (the ones in the leaves reached by the instance). From the union of all these response values, quantiles can be estimated. PIs can be computed from the quantiles, similar to what was done with quantile regression.

5. Experimental Validation

MOPSO and the two benchmark approaches (*QR* and *QRF*) were evaluated to construct PIs for the study case described in Section 2. The main comparison refers to using and not using the power measures at t_0. The latter is identified as $+P_{t_0}$. Therefor, six configurations were tested: *MOPSO*, $MOPSO + P_{t0}$, *QR*, $QR + P_{t0}$, *QRF*, and $QRF + P_{t0}$. PIs obtained from the different approaches were evaluated using two metrics: average interval width (AIW) and ratio (obtained as $PICP/AIW$). The larger is the ratio, the better is the trade-off between $PICP$ and AIW. Solutions that reach large $PICP$ with wide intervals are penalized with small ratio values.

PIs were obtained independently for each of the five forecast horizons (1–5 h). Several target nominal coverage values (*alpha*) were considered within the range 0.02–0.20. *QR* and *QRF* approaches must be run for each desired *alpha* value. The *MOPSO* approach is able to obtain PIs for all *alphas* in a single run (see Section 3). Ninety runs were executed for *MOPSO* (each run was valid for all the *alpha* values) and *QRF* because they are stochastic methods. For *QR*, only one run for each *alpha* was executed, because it is not stochastic. The following three subsections describe the tuning of the hyper-parameters of the methods, the experimental results, and the statistical significance tests.

5.1. Hyper-Parameter Tuning

Two hyper-parameters were tuned for *MOPSO*: number of hidden neurons and number of iterations of PSO. The methodology used a training and validation set approach, where the validation set was used to compare and select the best hyper-parameter values. The values of hidden neurons explored were: 2, 4, 6, 8, 10, 15, 20, 30, and 50. The different values for the PSO iterations ranged from 1000 to 16,000 in steps of 1000. Ninety runs were carried out for each number of neurons and iterations, starting with different random seeds. Therefore, each of the 90 runs had its own hyper-parameter tuning process, which resulted in an optimal number of neurons and iterations for each run. The optimal hyperparameters were chosen using the validation set hypervolume ([21]). It is important to remark that this parameter optimization process was carried out independently for each of the five forecasting horizons. The 90-run average (and standard deviation) of the best hyper-parameters for each horizon and configuration (*MOPSO* and $MOPSO + P_{t_0}$) is displayed in Table 1.

It is observed that the number of hidden neurons depends on the horizon, and long horizons seem to require fewer hidden neurons than short horizons. This trend is not systematic, but it is true for all stations and horizons that more neurons are required for the first hour than for the fifth hour. A possible explanation is that, for longer horizons, predictions are more noisy, and simpler models are required to avoid overfitting. In any case, it shows that tuning the number of neurons separately for each horizon is useful. The number of iterations is always between 12,000 and 14,000, below the maximum value of 16,000, which shows that the iteration limit was appropriate. Nevertheless, we tried increasing this value for a few runs, but no significant changes in the Pareto fronts were observed.

Table 1. Best combination of parameters for each prediction horizon.

Station 1	MOPSO		$MOPSO + P_{t_0}$	
Horizon	**Neurons**	**Iterations**	**Neurons**	**Iterations**
1 h	$23.7_{(\pm 15.0)}$	$12889_{(\pm 3030)}$	$23.2_{(\pm 13.8)}$	$13922_{(\pm 2284)}$
2 h	$20.9_{(\pm 13.1)}$	$13644_{(\pm 2849)}$	$24.2_{(\pm 16.1)}$	$13533_{(\pm 2695)}$
3 h	$14.1_{(\pm 11.1)}$	$12622_{(\pm 3337)}$	$12.4_{(\pm 9.74)}$	$12900_{(\pm 3198)}$
4 h	$15.9_{(\pm 11.8)}$	$12244_{(\pm 3709)}$	$15.6_{(\pm 12.8)}$	$12833_{(\pm 3188)}$
5 h	$14.0_{(\pm 9.65)}$	$13322_{(\pm 3072)}$	$12.1_{(\pm 7.25)}$	$13633_{(\pm 2542)}$
Station 2	MOPSO		$MOPSO + P_{t_0}$	
Horizon	Neurons	Iterations	Neurons	Iterations
1 h	$21.2_{(\pm 13.0)}$	$12989_{(\pm 2696)}$	$19.6_{(\pm 13.3)}$	$13544_{(\pm 2995)}$
2 h	$22.2_{(\pm 14.7)}$	$13522_{(\pm 2837)}$	$22.7_{(\pm 15.1)}$	$13822_{(\pm 2401)}$
3 h	$18.7_{(\pm 13.9)}$	$13056_{(\pm 2977)}$	$15.2_{(\pm 12.1)}$	$12744_{(\pm 3248)}$
4 h	$14.7_{(\pm 10.8)}$	$12967_{(\pm 3303)}$	$14.2_{(\pm 9.75)}$	$12433_{(\pm 3250)}$
5 h	$11.8_{(\pm 9.29)}$	$12878_{(\pm 3395)}$	$13.2_{(\pm 10.1)}$	$12656_{(\pm 3529)}$
Station 3	MOPSO		$MOPSO + P_{t_0}$	
Horizon	Neurons	Iterations	Neurons	Iterations
1 h	$21.7_{(\pm 13.0)}$	$13544_{(\pm 2623)}$	$18.0_{(\pm 13.5)}$	$12756_{(\pm 3520)}$
2 h	$17.7_{(\pm 13.5)}$	$12689_{(\pm 3114)}$	$20.9_{(\pm 13.7)}$	$13044_{(\pm 3263)}$
3 h	$18.6_{(\pm 13.9)}$	$13056_{(\pm 3279)}$	$19.2_{(\pm 13.5)}$	$12789_{(\pm 3224)}$
4 h	$16.6_{(\pm 12.1)}$	$12422_{(\pm 3219)}$	$15.3_{(\pm 11.4)}$	$12911_{(\pm 3100)}$
5 h	$13.5_{(\pm 10.9)}$	$12744_{(\pm 3092)}$	$14.8_{(\pm 11.4)}$	$13433_{(\pm 2825)}$

For *QR*, no parameter tuning is required. However, the performance of RF method depends on two main parameters: nodesize (minimum number of samples in the tree leaves) and mtry (number of input attributes used en the trees). Therefore, for *QRF*, we also performed hyper-parameter tuning to select the most suitable configuration. The values explored for nodesize were 2, 5, 10, 20, 30, 40, 50, and 100 and for mtry 2, 4, 6, 8, 10, 12, and 14 (the last one only when past value of power ($+P_{t_0}$) was used as input). The measure used for selecting the best combination of parameters for *QRF* was the ratio in the validation set. The study of best parameters was also carried out for each prediction horizon, as for *MOPSO* approach. In this case, values of the parameters are very similar for the three stations, the five forecasting horizons, and the different target *alpha* values. Nodesize values are around 5.6 ± 0.86 with no past power and around 5.6 ± 0.80 when $+P_{t_0}$ was used. For mtry, the average values are around 11.5 ± 0.62 when no past power was used and 12.5 ± 0.78 when $+P_{t_0}$ was used (bearing in mind that, when past power was used, an additional attribute was added to the set of inputs).

5.2. Experimental Results

To have a first view of the performance of the different methods, Table 2 shows the mean of the ratio values for each method and each horizon, separately for Stations 1–3. Each ratio value is the mean of the ratios for all the *alpha* values (0.02, 0.05, 0.08, 0.1, 0.15, and 0.2).

The results show that $MOPSO + P_{t0}$ obtains better results than *MOPSO* for the nearest horizons (especially Horizons 1 and 2) and for the three locations. For the other methods (*QR* and *QRF*), the $+P_{t0}$ variant also behaves better than the original method, and this situation occurs for all the prediction horizons and all the stations. Comparing $MOPSO + P_{t0}$ with the other methods, we can observe that, for all horizons and stations, it obtains higher ratio values. When we compare $QR + P_{t0}$ and $QRF + P_{t0}$ methods, we can observe that there is not a clear predominance of any method. In some cases, the ratio values of $QRF + P_{t0}$ are better (Station 1), but, in other cases, it is the opposite, as in Station 3. Next, the results are analyzed in more detail.

Table 2. Means of ratios for all the *alpha* values, for each prediction horizon and each method (Stations 1–3).

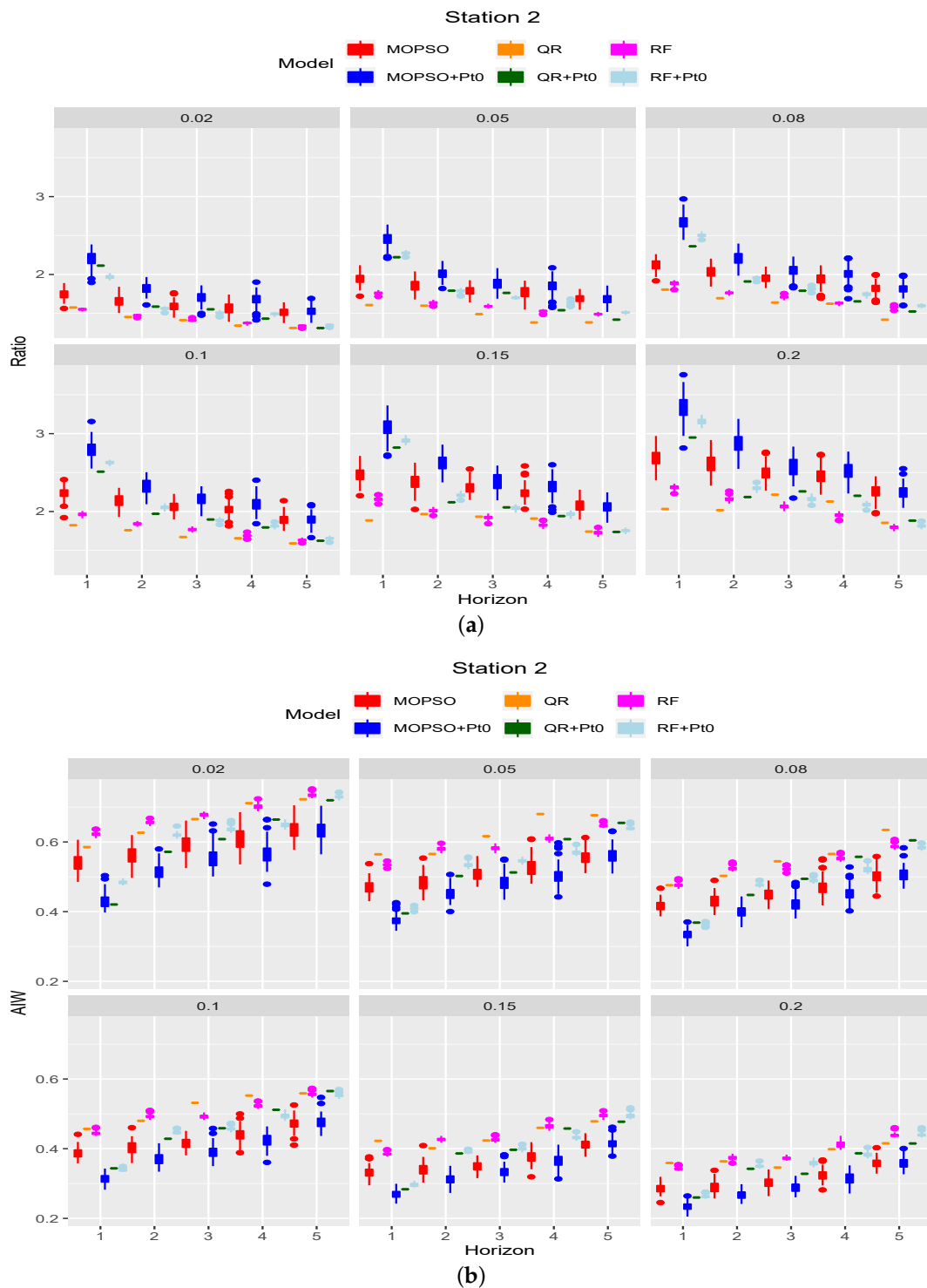|			Station 1			
Horizon	*MOPSO*	*MOPSO* + P_{t0}	*QR*	*QR* + P_{t0}	*QRF*	*QRF* + P_{t0}
1	2.07	2.69	1.71	2.47	1.80	2.58
2	2.23	2.30	1.80	1.98	1.88	1.96
3	2.35	2.34	1.90	1.91	1.97	1.98
4	2.37	2.36	1.92	1.95	2.01	2.01
5	2.33	2.33	2.04	2.08	2.16	2.18
			Station 2			
Horizon	*MOPSO*	*MOPSO* + P_{t0}	*QR*	*QR* + P_{t0}	*QRF*	*QRF* + P_{t0}
1	2.23	2.79	1.77	2.52	1.96	2.61
2	2.14	2.34	1.77	1.95	1.83	2.00
3	2.06	2.15	1.76	1.92	1.76	1.87
4	2.02	2.10	1.68	1.78	1.68	1.81
5	1.89	1.89	1.58	1.60	1.61	1.63
			Station 3			
Horizon	*MOPSO*	*MOPSO* + P_{t0}	*QR*	*QR* + P_{t0}	*QRF*	*QRF* + P_{t0}
1	2.37	2.89	2.02	2.77	2.09	2.84
2	2.30	2.41	2.02	2.20	1.97	2.14
3	2.15	2.20	1.96	2.01	1.90	1.95
4	2.13	2.14	1.92	1.95	1.85	1.89
5	2.11	2.09	1.90	1.91	1.86	1.87

Figures 2–4 display the results (ratio and *AIW*) for each of the three stations, respectively. Each of the figures contain two plots, top for the ratio (a) and bottom for *AIW* (b). These figures compare *MOPSO*, *QR*, *QRF*, *MOPSO* + P_{t0}, *QR* + P_{t0}, and *QRF* + P_{t0}. Information for *MOPSO*, *MOPSO* + P_{t0}, *QRF*, and *QRF* + P_{t0} is displayed by means of box plots, because they are a good summary of the 90 runs. *QR* and *QR* + P_{t0} are shown as horizontal lines, because *QR* is not stochastic. Plots can be seen for some representative *alpha* values (0.02, 0.05, 0.08, 0.1, 0.15, and 0.2). Figure 2a displays the ratio of the six methods tested for Station 1. It is readily apparent that ratio of *MOPSO* + P_{t0} is larger than *MOPSO* for the first horizon, and, to a lesser degree, for the second one. This should be expected, because the influence of measures at time t_0 should decrease as the horizon is farther away from t_0. It can also be noticed that the difference between *MOPSO* + P_{t0} and *MOPSO* decreases as *alpha* increases. For instance, for the second horizon, *MOPSO* + P_{t0} is clearly better than *MOPSO* for *alpha* = 0.02 but quite similar for *alpha* = 0.2. The second issue to notice in Figure 2 is that, although *QR* and *QRF* also benefit from using t_0 at short horizons, *MOPSO* configurations typically outperform their *QR* and *QRF* counterparts (i.e., *MOPSO* + P_{t0}'s ratio is larger than those of *QR* + P_{t0} and *QRF* + P_{t0}, and it also is when t_0 is not used). This is true for all horizons and *alpha*s. The *AIW* plot on the bottom of Figure 2b confirms the previous findings: *MOPSO* + P_{t0} intervals are narrower than *MOPSO* for the first two horizons (and that is also true for *QR*, *QR* + P_{t0}, *QRF*, and *QRF* + P_{t0}), highlighting the importance of using t_0 for obtaining narrow intervals. Again, *MOPSO* configurations always obtain narrower intervals than the *QR* and *QRF* baselines. Figure 3 supports the previous results for Station 2. In this case, higher ratios are obtained by *MOPSO* + P_{t0} (vs. *MOPSO*, *QR* + P_{t0} and *QRF* + P_{t0}) for longer horizons (from 1 to 4 h). Finally, those results are mirrored by Station 3, as shown in Figure 4, although in this case there is an anomaly for *alpha* = 0.1, where *QR* + P_{t0} manages to obtain better ratios (and narrower intervals) than *MOPSO* + P_{t0} for the first forecasting horizon. In this station, it is also observed that *QRF* + P_{t0} obtains similar ratios as *MOPSO* + P_{t0} for *alpha* > 0.05 and Horizon 1. For the remaining horizons, *MOPSO* + P_{t0} outperforms *QRF* + P_{t0}.
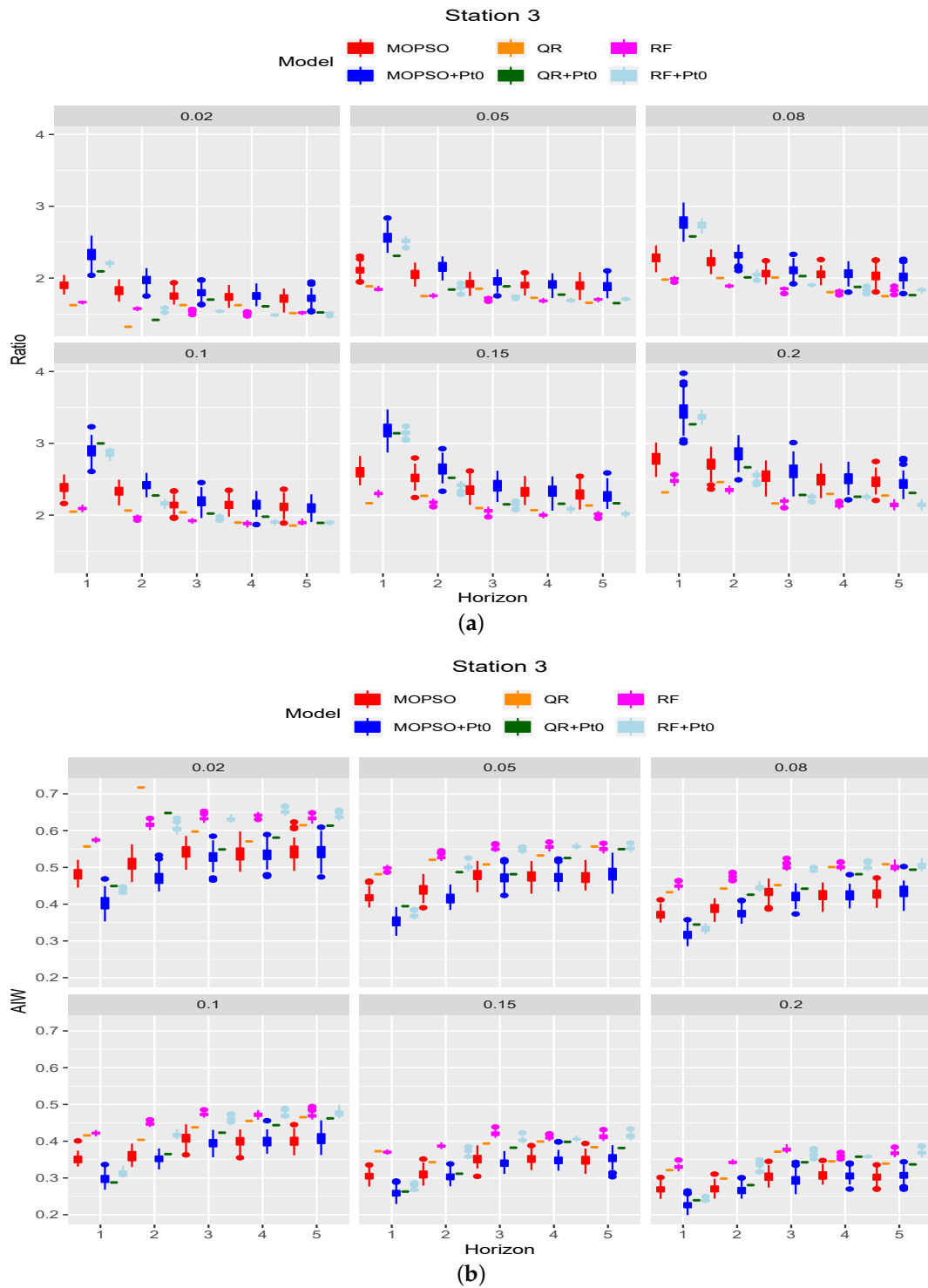
(**a**)

(**b**)
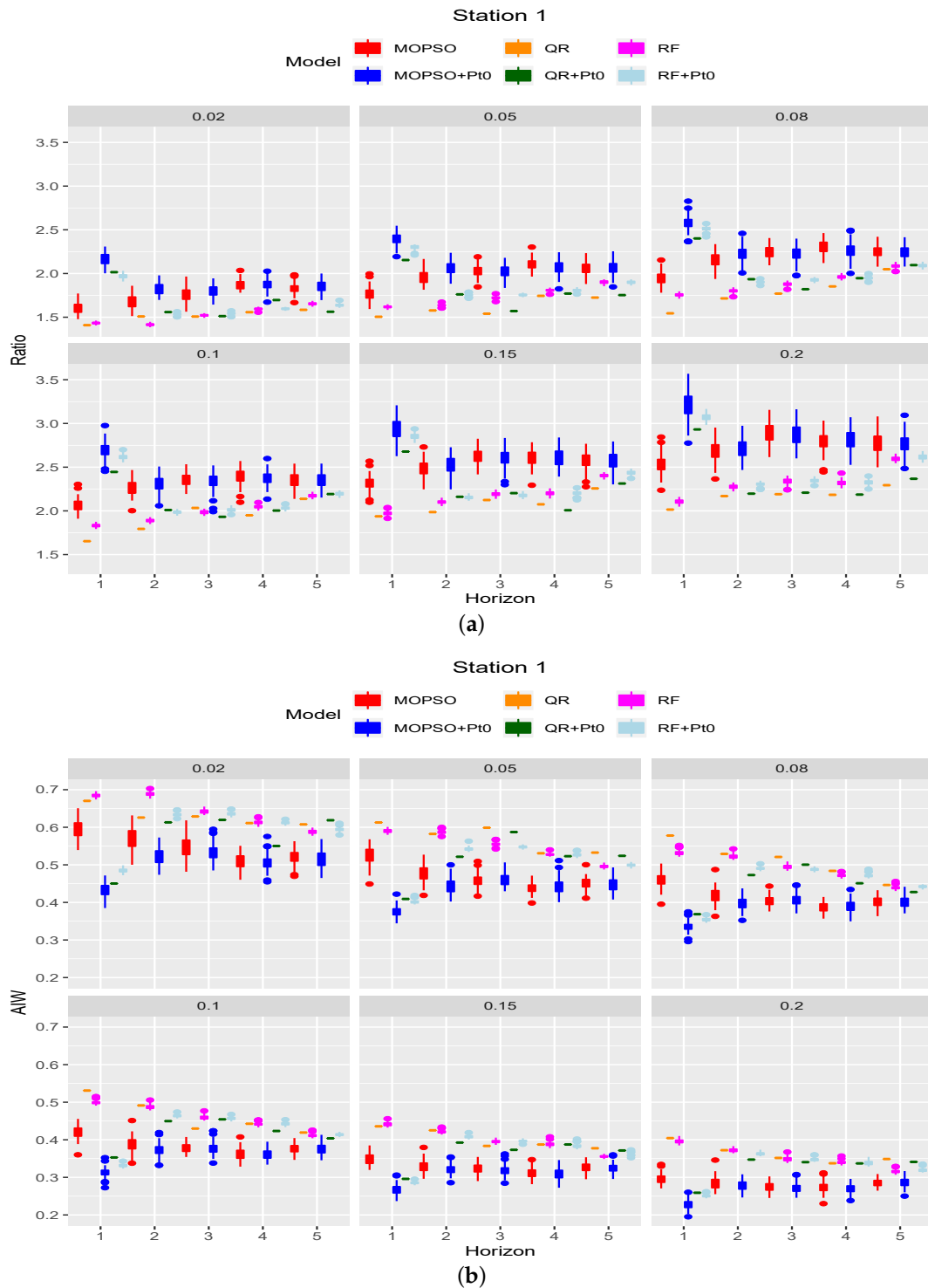
Figure 2. Station 1: (**a**) *PICP/AIW* ratio per prediction horizon; and (**b**) *AIW* per prediction horizon. *Alpha* values = 0.02 to 0.2.

To show the PIs constructed for some specific days, Figure 5 displays the PIs obtained by *MOPSO*, *MOPSO + Pt0*, and *QRF + Pt0* on 2013-02-25, 2014-03-19, 2014-04-19, 2013-05-18, and 2012-06-26. PIs were obtained for Station 2 and *alpha* = 0.1. Each row in Figure 5 corresponds to each different day. The left column compares *MOPSO + Pt0* (grey) with *MOPSO* (yellow) and the right column compares *MOPSO + Pt0* (grey) with *QRF + Pt0* (yellow). The x-axis displays the five forecasting horizons and the y-axis shows the PIs (in grey and yellow) and the actual normalized power (a black line). Although the

results depend on the particular day, in general, it is observed that $MOPSO + P_{t0}$ provides narrower intervals than $MOPSO$ (except in the last horizon (5 h) for some of the days) and $QRF + P_{t0}$.

Figure 3. Station 2: (**a**) $PICP/AIW$ ratio per prediction horizon; and (**b**) AIW per prediction horizon. *Alpha* values = 0.02 to 0.2.

Figure 4. Station 3: (**a**) *PICP/AIW* ratio per prediction horizon; and (**b**) *AIW* per prediction horizon. *Alpha* values = 0.02 to 0.2.
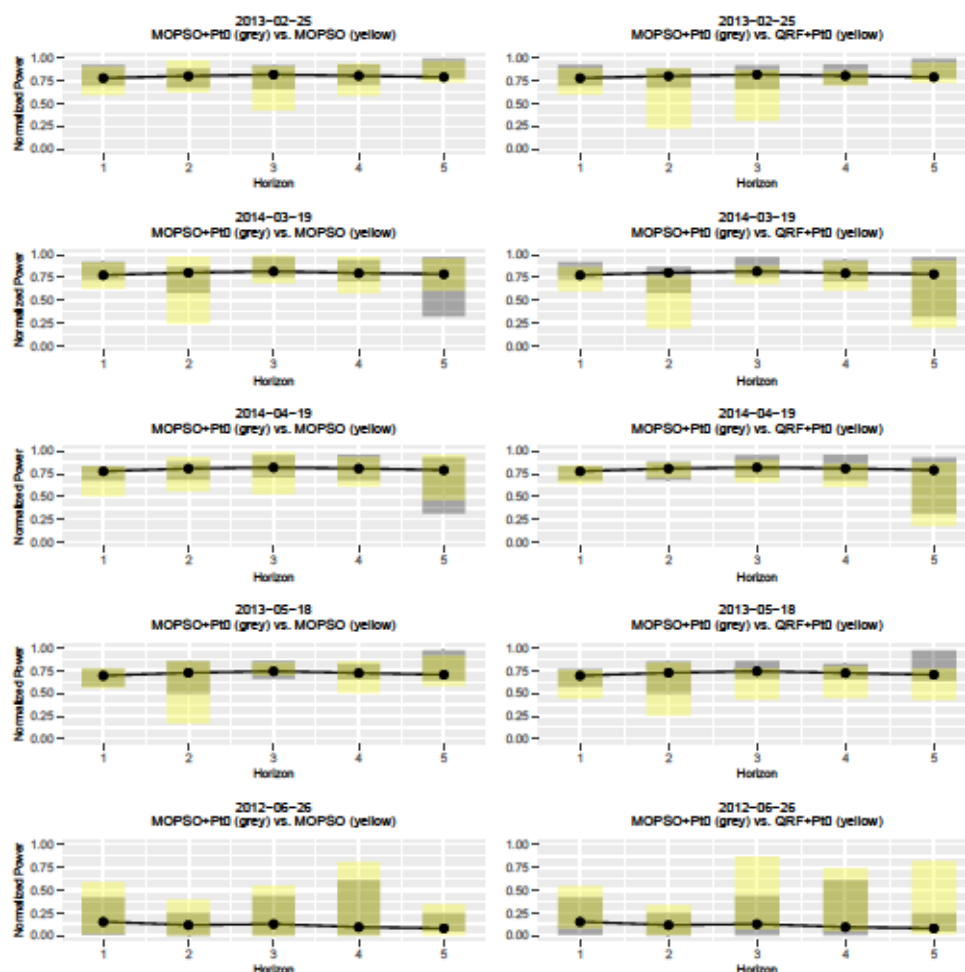
Figure 5. PIs for the five forecast horizons for four specific days of the year (one day per row): The left column compares $MOPSO + P_{t0}$ (grey) with $MOPSO$ (yellow) and the right column compares $MOPSO + P_{t0}$ (grey) with $QRF + P_{t0}$ (yellow).

### 5.3. Statistical Significant Tests

The above figures offer a qualitative view of the results, but, given that $MOPSO$ and QRF were run 90 times, statistical tests could also be computed. In this work, we used the Wilcoxon test. The results are summarized in Figures 6–8, for Stations 1–3, repectively. In this case, only tests for the ratio are shown (but AIW follows a similar pattern and can be found in Appendix A. Part (a) (top) of Figures 6–8 compares $MOPSO + P_{t0}$ and $MOPSO$. A blue point is used when the $MOPSO + P_{t0}$'s ratio is significantly larger than $MOPSO$, and red for the other way around. Color grey signals that $MOPSO + P_{t0}$ and $MOPSO$ are not statistically different. lThe lower plots (b) and (c) of Figures 6–8 perform the same comparison, but for $MOPSO + P_{t0}$ versus $QR + P_{t0}$ and for $MOPSO + P_{t0}$ versus $QRF + P_{t0}$. Results for Station 1 (see Figure 6a) show that, indeed, for the first two horizons, $MOPSO + P_{t0}$'s ratios are better than $MOPSO$'s (blue points for all *alphas*). However, beyond the second hour, $MOPSO + P_{t0}$ is not necessarily significantly better (and, in very few cases, it is significantly worse: red points). The comparisons between $MOPSO + P_{t0}$ and $QR + P_{t0}$ and $MOPSO + P_{t0}$ and $QRF + P_{t0}$ offer more systematic results: they are almost all blue points, which means that $MOPSO + P_{t0}$ is basically significantly better than $QR + P_{t0}$ and better than $QRF + P_{t0}$. The second station shows even cleaner results: Figure 7a displays blue points for the first four hours, and mostly grey for the fifth horizon. Again, the blue-filled plot of Figure 7b means that $MOPSO + P_{t0}$ is significantly better than $QR + P_{t0}$ and $QRF + P_{t0}$ in all cases. In summary, for Station 2, t_0 has a beneficial effect on $MOPSO$ for even longer horizon spans (up to the fourth hour). Finally, Figure 8

plots results for Station 3. In this case, the top plot shows blue points up to the third hour, all grey for the fourth horizon, but mostly red for the fifth one. With respect to $MOPSO + P_{t0}$ vs. $QR + P_{t0}$ (Figure 8b), it is again mostly blue points, but the anomalies mentioned in the qualitative results pop up again here (red points), for the first horizon and $\alpha = 0.09$, 0.1, and 0.11. In this figure it is also observed than the observed similarity of $QRF + P_{t0}$ and $MOPSO + P_{t0}$ previously observed for Station 3 and for the first horizon manifests as differences not statistically significant (grey points).
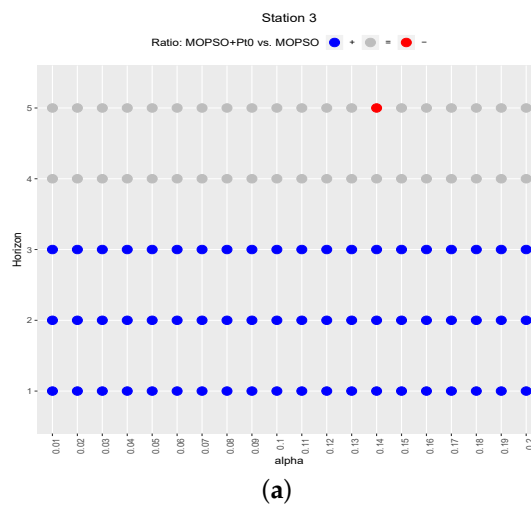

(a)

(b)

(c)
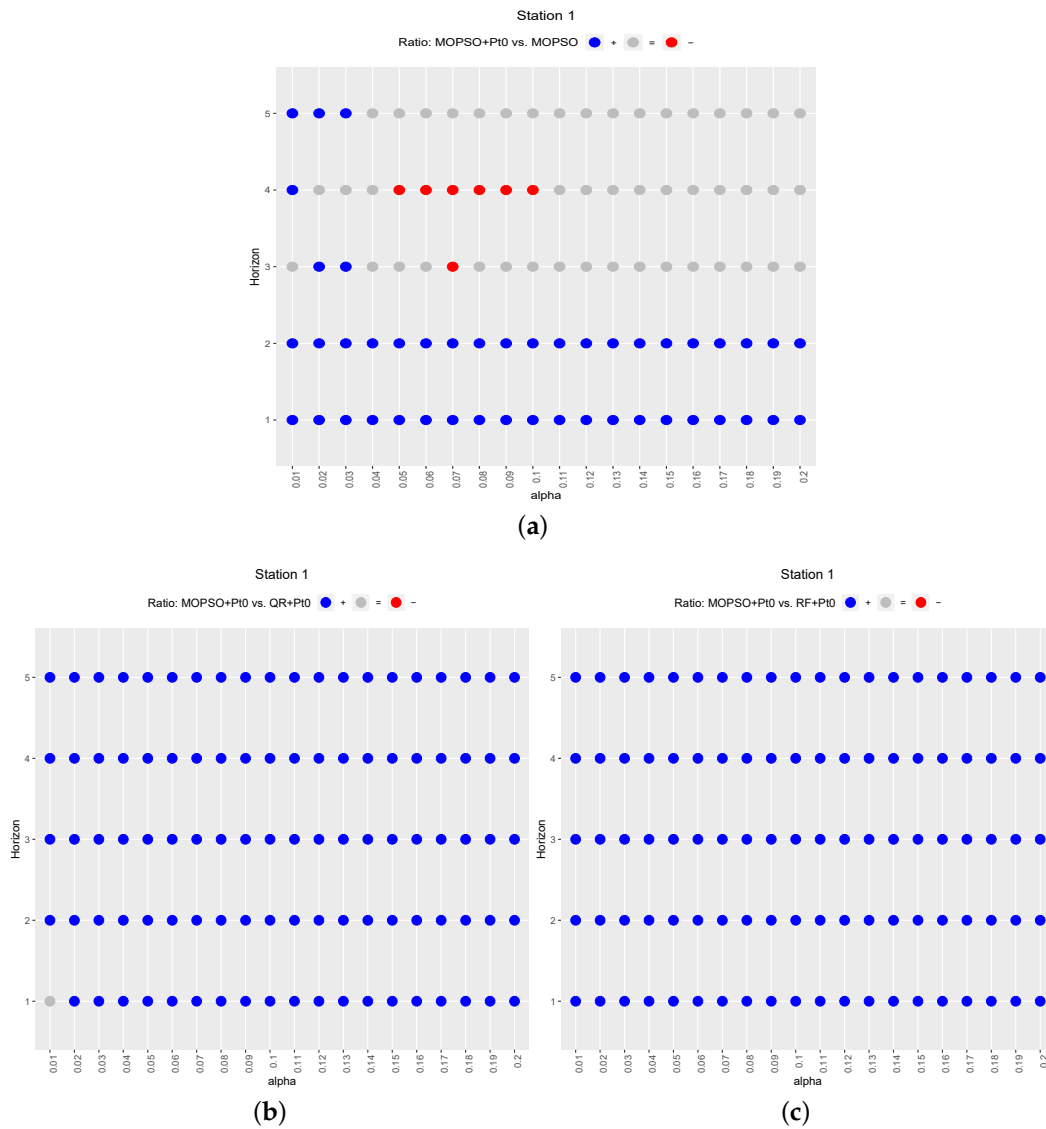
Figure 6. Station 1. Statistical tests for ratio: (**a**) $MOPSO + P_{t0}$ vs. $MOPSO$; (**b**) $MOPSO + P_{t0}$ vs. $QR + P_{t0}$; and (**c**) $MOPSO + P_{t0}$ vs. $QRF + P_{t0}$.
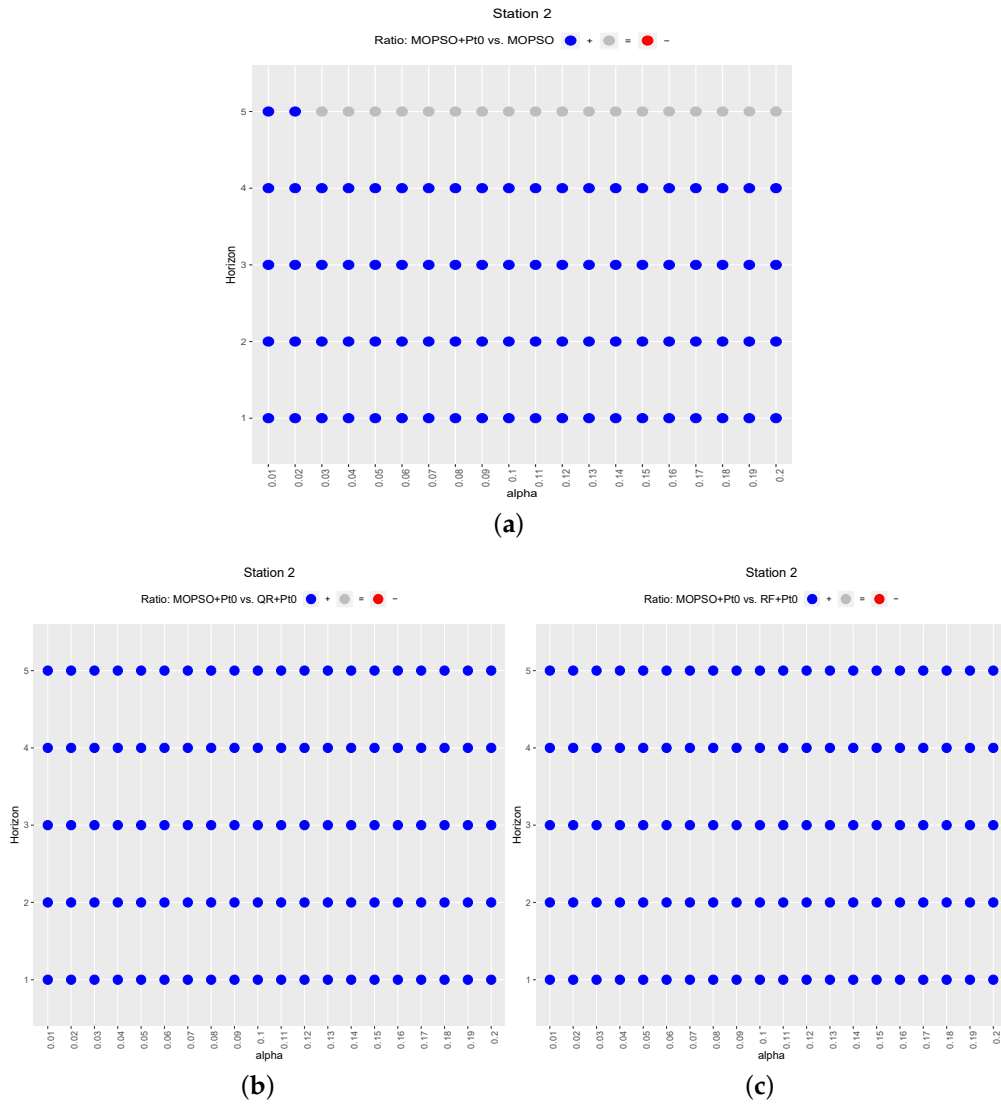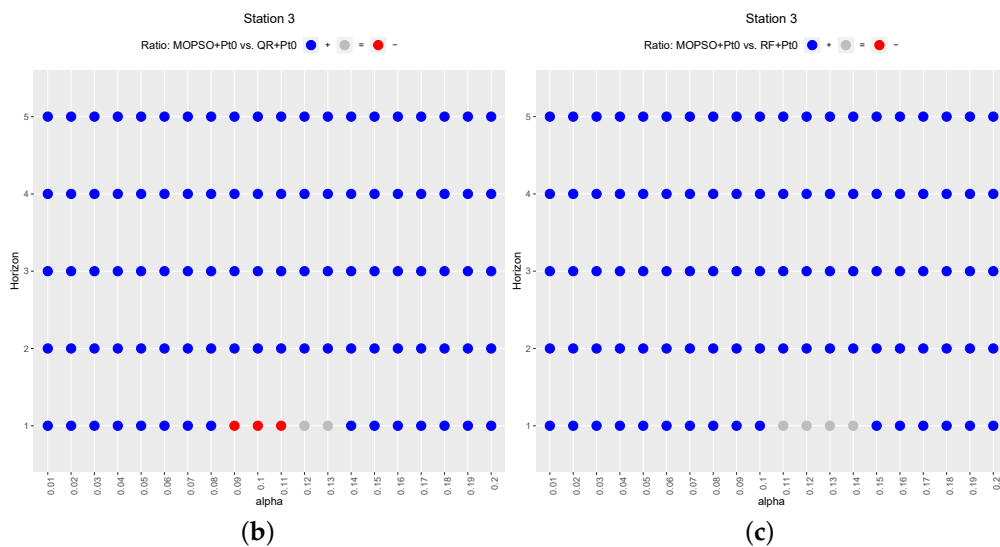
Figure 7. Station 2. Statistical tests for ratio: (**a**) $MOPSO + P_{t0}$ vs. $MOPSO$; (**b**) $MOPSO + P_{t0}$ vs. $QR + P_{t0}$; and (**c**) $MOPSO + P_{t0}$ vs. $QRF + P_{t0}$.

Figure 8. *Cont.*

Figure 8. Station 3. Statistical tests for ratio: (**a**) $MOPSO + P_{t0}$ vs. $MOPSO$; (**b**) $MOPSO + P_{t0}$ vs. $QR + P_{t0}$; and (**c**) $MOPSO + P_{t0}$ vs. $QRF + P_{t0}$.

In summary, $MOPSO + P_{t0}$ is significantly better than $MOPSO$ for the short horizons for all *alpha* values. For long horizons, $MOPSO + P_{t0}$ is not significantly worse than $MOPSO$ except for some *alpha* values (mostly in Station 1). $MOPSO + P_{t0}$ is significantly better than $QR + P_{t0}$ and $QRF + P_{t0}$ for all horizons, except for a few *alpha* values for the first horizon in Station 3.

6. Conclusions

The LUBE approach is an interesting alternative for estimating PIs in the context of probabilistic forecasting because it is able to estimate directly the lower and upper bounds of PIs. LUBE can be optimized by $MOPSO$, a multi-objective evolutionary technique, so that neural networks can be trained to simultaneously optimize the two conflicting properties of PIs: coverage and width. The result of this optimization is a Pareto front, from which solutions can be extracted according to the desired coverage value. In this study, this approach was used to address two issues to obtain PIs for solar power forecasting. The first one was to estimate PIs for five intra-day forecasting horizons from 1 to 5 h. The second, and more important one, was to study the influence of using measured solar power at time t_0 on the quality of PIs. Previous work has shown that this is useful for improving point forecasting, but we studied this issue in the context of probabilistic forecasting.

The approach was applied to data from three solar stations in Australia and experiments were carried out using two main configurations: (1) using meteorological forecasts variables as inputs to the methods; and (2) using additionally the measured solar output power as input. To analyze how far the influence of measured output reaches, hourly forecasts horizons from 1 to 5 h were used. The quality of prediction intervals was estimated using the coverage/width ratio and the width of the intervals. High values for ratio mean that intervals have a good trade-off between coverage and width. This study was done for several desired coverage values (or *alpha*) from 0.01 to 0.20.

The results show that the ratio is improved by using the measured additional information for the two first horizons on the three locations studied and all the desired coverage values. However, although for one of the station this beneficial influence reaches up to the fourth horizon, in none of the stations the ratio is improved for the farthest horizon tested (5 h). The same trend can be observed for interval width. Experiments were replicated 90 times for different random seeds and statistical significance tests were performed, which show that the mentioned results are statistically significant. Thus, it can be concluded that using measured solar power reduces the uncertainty of the intervals for short forecasting horizons.

The *MOPSO* approach was compared with *QR* and *QRF* as baseline methods. Both were tested in the same conditions as *MOPSO*, with and without the measured power at time t_0. For all approaches, the use of measured power helped to obtain better PIs, especially in the first horizons. The comparison also shows that $MOPSO + P_{t0}$ is significantly better in all cases except a few *alpha* values in one of the stations.
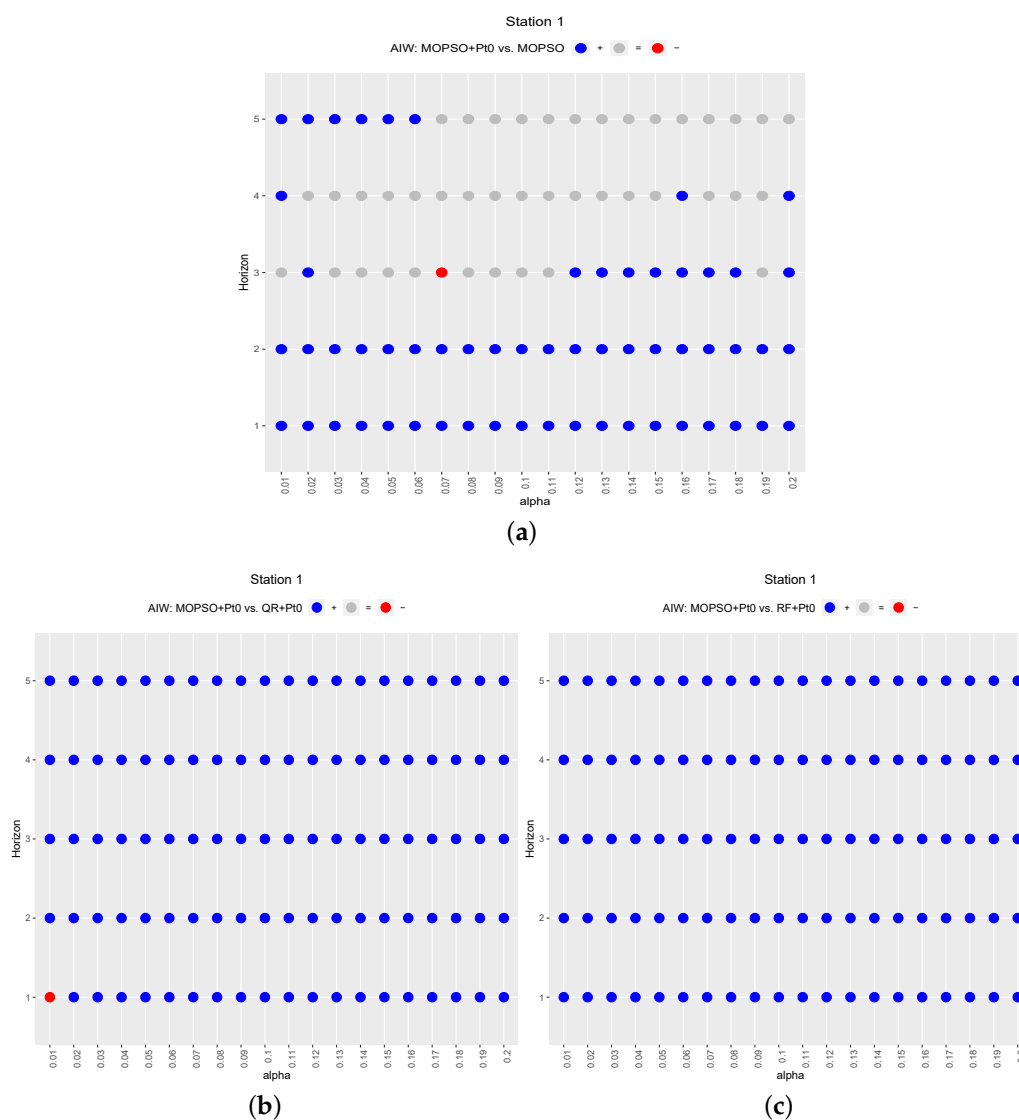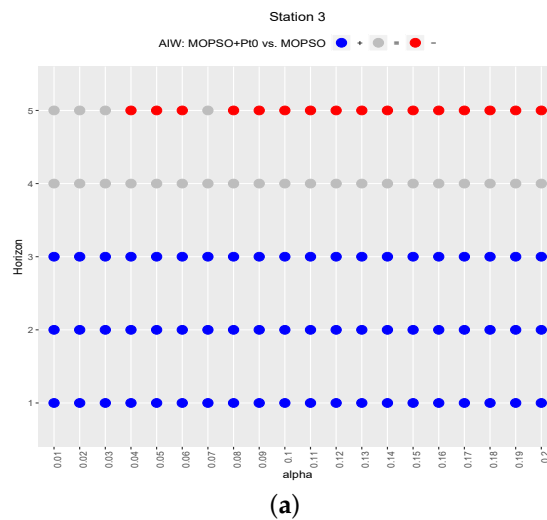
Appendix A

This appendix displays the statistical results for *AIW*.

Figure A1. Station 1. Statistical tests for wideness: (**a**) $MOPSO + P_{t0}$ vs. *MOPSO*; (**b**) $MOPSO + P_{t0}$ vs. $QR + P_{t0}$; and (**c**) $MOPSO + P_{t0}$ vs. $QRF + P_{t0}$.

(**a**)

(**b**)   (**c**)
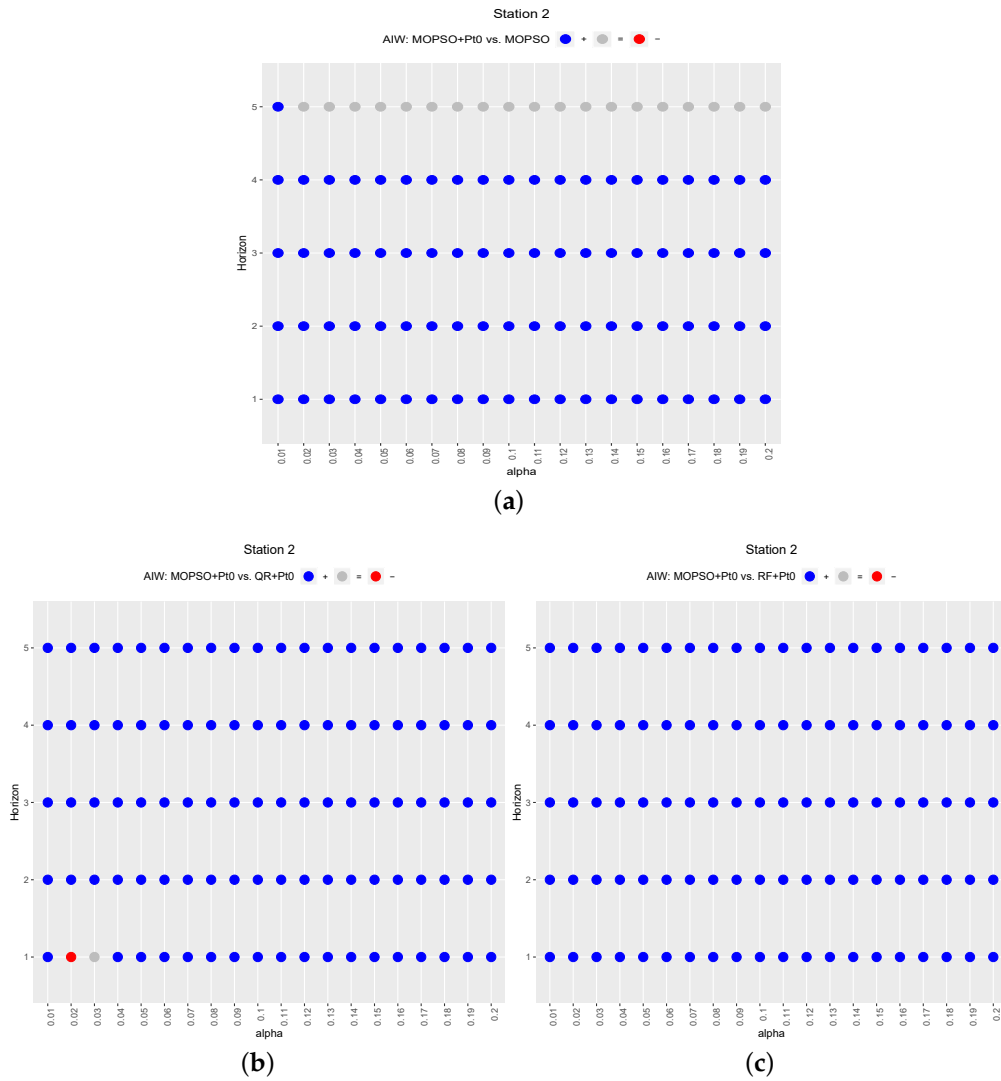
Figure A2. Station 2. Statistical tests for wideness: (**a**) $MOPSO + P_{t0}$ vs. $MOPSO$; (**b**) $MOPSO + P_{t0}$ vs. $QR + P_{t0}$; and (**c**) $MOPSO + P_{t0}$ vs. $QRF + P_{t0}$.
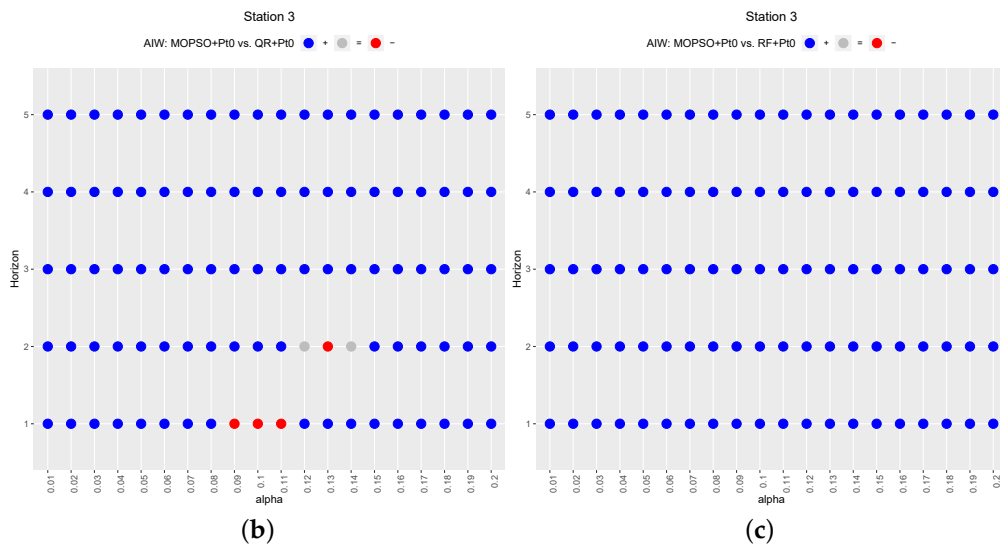

(**a**)

Figure A3. *Cont.*

Figure A3. Station 3. Statistical tests for wideness: (**a**) $MOPSO + P_{t0}$ vs. $MOPSO$; (**b**) $MOPSO + P_{t0}$ vs. $QR + P_{t0}$; and (**c**) $MOPSO + P_{t0}$ vs. $QRF + P_{t0}$.

References

1. Raza, M.Q.; Nadarajah, M.; Ekanayake, C. On recent advances in PV output power forecast. *Sol. Energy* **2016**, *136*, 125–144. [CrossRef]

2. Pinson, P.; Chevallier, C.; Kariniotakis, G.N. Trading wind generation from short-term probabilistic forecasts of wind power. *IEEE Trans. Power Syst.* **2007**, *22*, 1148–1156. [CrossRef]

3. Matos, M.A.; Bessa, R.J. Setting the operating reserve using probabilistic wind power forecasts. *IEEE Trans. Power Syst.* **2010**, *26*, 594–603. [CrossRef]

4. Van der Meer, D.W.; Widén, J.; Munkhammar, J. Review on probabilistic forecasting of photovoltaic power production and electricity consumption. *Renew. Sustain. Energy Rev.* **2018**, *81*, 1484–1512. [CrossRef]

5. Bessa, R.; Möhrlen, C.; Fundel, V.; Siefert, M.; Browell, J.; Haglund El Gaidi, S.; Hodge, B.M.; Cali, U.; Kariniotakis, G. Towards improved understanding of the applicability of uncertainty forecasts in the electric power industry. *Energies* **2017**, *10*, 1402. [CrossRef]

6. Dobschinski, J.; Bessa, R.; Du, P.; Geisler, K.; Haupt, S.E.; Lange, M.; Möhrlen, C.; Nakafuji, D.; de la Torre Rodriguez, M. Uncertainty forecasting in a nutshell: Prediction models designed to prevent significant errors. *IEEE Power Energy Mag.* **2017**, *15*, 40–49. [CrossRef]

7. Koenker, R.; Hallock, K.F. Quantile regression. *J. Econ. Perspect.* **2001**, *15*, 143–156. [CrossRef]

8. Cannon, A.J. Quantile regression neural networks: Implementation in R and application to precipitation downscaling. *Comput. Geosci.* **2011**, *37*, 1277–1284. [CrossRef]

9. Friedman, J.H. Greedy Function Approximation: A Gradient Boosting Machine. *Ann. Stat.* **2000**, *29*, 1189–1232. [CrossRef]

10. Friedman, J.H. Stochastic gradient boosting. *Comput. Stat. Data Anal.* **2002**, *38*, 367–378. [CrossRef]

11. Meinshausen, N. Quantile regression forests. *J. Mach. Learn. Res.* **2006**, *7*, 983–999.

12. Delle Monache, L.; Eckel, F.A.; Rife, D.L.; Nagarajan, B.; Searight, K. Probabilistic weather prediction with an analog ensemble. *Mon. Weather Rev.* **2013**, *141*, 3498–3516. [CrossRef]

13. Cervone, G.; Haupt, S.E. Probabilistic Forecasting of solar energy generation using numerical modeling and remote sensing. In Proceedings of the AGU Fall Meeting Abstracts, San Francisco, CA, USA, 12–16 December 2018.

14. Clemente-Harding, L.; Young, G.; Cervone, G.; Delle Monache, L.; Haupt, S.E. Toward a Search Space Extension for the Analog Ensemble technique to Improve Probabilistic Predictions. In Proceedings of the AGU Fall Meeting Abstracts, San Francisco, CA, USA, 12–16 December 2018.

15. Pinson, P.; Nielsen, H.A.; Møller, J.K.; Madsen, H.; Kariniotakis, G.N. Non-parametric probabilistic forecasts of wind power: Required properties and evaluation. *Wind Energy* **2007**, *10*, 497–516. [CrossRef]

16. Khosravi, A.; Nahavandi, S.; Creighton, D.; Atiya, A.F. Lower upper bound estimation method for construction of neural network-based prediction intervals. *IEEE Trans. Neural Netw.* **2011**, *22*, 337–346. [CrossRef]

17. Wan, C.; Xu, Z.; Pinson, P. Direct interval forecasting of wind power. *Power Syst. IEEE Trans.* **2013**, *28*, 4877–4878. [CrossRef]

18. Khosravi, A.; Nahavandi, S. Combined nonparametric prediction intervals for wind power generation. *Sustain. Energy IEEE Trans.* **2013**, *4*, 849–856. [CrossRef]

19. Kirkpatrick, S.; Gelatt, C.D.; Vecchi, M.P. Optimization by simulated annealing. *Science* **1983**, *220*, 671–680. [CrossRef]

20. Eberhart, R.C.; Shi, Y.; Kennedy, J. *Swarm Intelligence*; Elsevier: Amsterdam, The Netherlands, 2001.

21. Galván, I.M.; Valls, J.M.; Cervantes, A.; Aler, R. Multi-objective evolutionary optimization of prediction intervals for solar energy forecasting with neural networks. *Inf. Sci.* **2017**, *418*, 363–382. [CrossRef]

22. McGovern, A.; Gagne, D.J.; Basara, J.; Hamill, T.M.; Margolin, D. Solar energy prediction: An international contest to initiate interdisciplinary research on compelling meteorological problems. *Bull. Am. Meteorol. Soc.* **2015**, *96*, 1388–1395. [CrossRef]

23. Aguiar, L.M.; Pereira, B.; Lauret, P.; Díaz, F.; David, M. Combining solar irradiance measurements, satellite-derived data and a numerical weather prediction model to improve intra-day solar forecasting. *Renew. Energy* **2016**, *97*, 599–610. [CrossRef]

24. Wolff, B.; Kühnert, J.; Lorenz, E.; Kramer, O.; Heinemann, D. Comparing support vector regression for PV power forecasting to a physical modeling approach using measurement, numerical weather prediction, and cloud motion data. *Sol. Energy* **2016**, *135*, 197–208. [CrossRef]

25. Martín-Vázquez, R.; Aler, R.; Galván, I. Wind Energy Forecasting at Different Time Horizons with Individual and Global Models. In Proceedings of the IFIP International Conference on Artificial Intelligence Applications and Innovations, lRhodes, Greece, 25–27 May 2018; Springer: New York, NY, USA, 2018; pp. 240–248.

26. Martín-Vázquez, R.; Huertas-Tato, J.; Aler, R.; Galván, I. Studying the Effect of Measured Solar Power on Evolutionary Multi-objective Prediction Intervals. In Proceedings of the International Conference on Intelligent Data Engineering and Automated Learning, Madrid, Spain, 21–23 November 2018; Springer: New York, NY, USA, 2018; pp. 155–162.

27. Lauret, P.; David, M.; Pedro, H. Probabilistic solar forecasting using quantile regression models. *Energies* **2017**, *10*, 1591. [CrossRef]

28. Zamo, M.; Mestre, O.; Arbogast, P.; Pannekoucke, O. A benchmark of statistical regression methods for short-term forecasting of photovoltaic electricity production. Part II: Probabilistic forecast of daily production. *Sol. Energy* **2014**, *105*, 804–816. [CrossRef]

29. Abuella, M.; Chowdhury, B. Hourly probabilistic forecasting of solar power. In Proceedings of the IEEE 2017 North American Power Symposium (NAPS), Morgantown, WV, USA, 17–19 September 2017; pp. 1–5.

30. Hong, T.; Pinson, P.; Fan, S.; Zareipour, H.; Troccoli, A.; Hyndman, R.J. Probabilistic energy forecasting: Global Energy Forecasting Competition 2014 and beyond. *Int. J. Forecast.* **2016**, *32*, 896–913. [CrossRef]

31. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]