

# The Combined Effect of Applying Feature Selection and Parameter Optimization on Machine Learning Techniques for Solar Power Prediction

Md Rahat Hossain\*, Amanullah Maung Than Oo, A B M Shawkat Ali

Power Engineering Research Group (PERG), Central Queensland University, Rockhampton, Australia

\*Corresponding author: [m.hossain@cqu.edu.au](mailto:m.hossain@cqu.edu.au)

Received December 24, 2012; Revised January 29, 2013; Accepted February 26, 2013

**Abstract** This paper empirically shows that the combined effect of applying the selected feature subsets and optimized parameters on machine learning techniques significantly improves the accuracy for solar power prediction. To provide evidence, experiments are carried on in two phases. For all the experiments the machine learning techniques namely Least Median Square (LMS), Multilayer Perceptron (MLP) and Support Vector Machine (SVM) are used. In the first phase five well-known wrapper feature selection methods are used to obtain the prediction accuracy of machine learning techniques with selected feature subsets and default parameter settings. The experiments from the first phase demonstrate that holding the default parameters, LMS, MLP and SVM provides better prediction accuracy (i.e. reduced MAE and MASE) with selected feature subsets rather than without selected feature subsets. After getting improved prediction accuracy from the first phase, the second phase continues the experiments to optimize machine learning parameters and the prediction accuracy of those machine learning techniques are re-evaluated through adopting both the optimized parameter settings and selected feature subsets. The comparison between the results of two phases clearly shows that the later phase (i.e. machine learning techniques with selected feature subsets and optimized parameters) provides substantial improvement in the accuracy for solar power prediction than the earlier phase (i.e. machine learning techniques with selected feature subsets and default parameters). Experiments are carried out using reliable and real life historical meteorological data. The machine learning accuracy of solar radiation prediction is justified in terms of statistical error measurement and validation metrics. Experimental results of this paper facilitate to make a concrete verdict that providing more attention and effort towards the feature subset selection and machine learning parameter optimization (e.g. combined effect of selected feature subsets and optimized parameters on prediction accuracy which is investigated in this paper) can significantly contribute to improve the accuracy of solar power prediction.

**Keywords:** *feature selection, machine learning, regression algorithm, solar radiation, parameter optimization, DTREG*

## 1. Introduction

Feature selection can be considered one of the main pre-processing steps of machine learning [1]. It contributes considerably by the reduction of dimension as well as eliminating inappropriate data. It is quite capable to improve learning accuracy in computational intelligence. The feature selection aspect is fairly significant for the reason that with the same training data it may happen that individual regression algorithm can perform better with different feature sub sets [2]. The success of machine learning on a particular task is affected by many factors. Among those factors first and foremost is the representation and quality of the instance data [3]. The training stage becomes critical with the existence of noisy, irrelevant and redundant data. Sometimes the real life data contain too much information among those very little is useful for desired purpose. Therefore, it is not

important to include every piece of information from the raw data source for modelling.

All the algorithms to perform feature selection consist of two common aspects. One is the search method which is actually a selection algorithm to generate designed feature subsets and attempts to reach the most advantageous. Another aspect is called evaluator which is basically an evaluation algorithm to make a decision about the goodness of the planned feature subset and finally returns the assessment about righteousness of the search method [4]. On the other hand, lacking of an appropriate stopping condition the feature selection procedure could run exhaustively or everlastingly all the way throughout raw data set. It may be discontinued whenever any attribute is inserted or deleted but ultimately not producing a better subset or whenever a subset is produced which provides the maximum benefits according to some assessing functions. A feature selector may stop manipulating features when the merit of a current feature subset stops improving or conversely does not degrade.

Based on some evaluation functions and calculations feature selection methods find out the best feature from different candidate subsets. Usually feature selection methods are classified into two general groups (i.e. filter and wrapper) [5]. Inductive algorithms are used by wrapper methods as the evaluation function whereas filter methods are independent of the inductive algorithm. Wrapper methods work along wrapping the feature selection in conjunction with the induction algorithm to be used and to accomplish it wrapper methods use cross-validation.

Parameter tuning or optimization plays a fundamental role in machine learning techniques [6]. To achieve good quality generalization, it is essential to select an adequately good model parameter set for the particular learning problem. The selection of model parameters can radically influence the excellence of the solution. Poor selection of parameters can effect in failure of the method [7]. The parameter optimization is a procedure of tuning the learning parameters (e.g. number of neurons in the hidden layer for neural networks, or the kernel selection for support vector machine), of the machine learning techniques to deal with specific type of problems. In spite of the reasonable default parameter settings of the machine learning algorithms, it is not guaranteed that those will be optimal for any specific problem [8]. The most important idea of parameter tuning is to select a subset of significant parameters to build robust learning models. From the theoretical viewpoint, it can be shown that optimal parameter selection demands a meticulous search of all potential subsets of parameters. This is unrealistic whenever a great number of parameters are on hand. Therefore in the field of machine learning the challenge is to find out an acceptable set of parameters rather than an optimal parameter set [6]. Eventually the satisfactory parameter set is considered as an optimal set. The methods or procedure to optimize a system largely varies and depends on the purpose of that system involved but the aim of all optimization matches to a common interest; to achieve the optimal outcome. Parameters are either set by common, non-task-specific rules (e.g. hand-tuning) or they are automatically tuned by predictive modeling software [9]. However, some published research work for solar power prediction deals with either feature selection or algorithm parameter optimization in very limited scope; the effectiveness of using both of them at the same time is not so far methodically investigated.

Widely used wrapper selection methods are briefly discussed in the following section. Methods of parameter optimization (e.g. hand-tuned and auto parameter optimization by predictive modelling program) are briefly illustrated in section 3. Section 4 deals with real life data collection and analysis of the data set. Section 5 handles the first phase of the experiments that is selecting potential feature subsets by wrapper selection methods and applying them on machine learning techniques with default parameter settings. The results obtained from this phase are compared with the prediction results of machine learning techniques without applying feature selection (WAFS) methods to observe the improvement. Section 6 and 7 conducts the second phase of the experiments that is optimizing machine learning parameters and the prediction accuracy of those machine learning techniques are re-evaluated through adopting both the optimized

parameter settings and selected feature subsets. The evaluation between the results of two phases evidently shows that the later phase (i.e. machine learning techniques with selected feature subsets and optimized parameters) provides considerable improvement in the accuracy for solar power prediction than the earlier phase (i.e. machine learning techniques with selected feature subsets and default parameters). Concluding remarks are provided in final section of this paper.

## 2. Wrapper Methods of Feature Selection

The wrapper methods use the performance (e.g. regression, classification or prediction accuracy) of an induction algorithm for feature subset evaluation. The concept of wrapper approach is presented in the Figure 1 [10]. Wrapper methods evaluate the goodness of each selected feature subset by applying that induction algorithm to the original dataset using the selected features in the subset. Usually wrapper methods are able to generate potential feature subsets with high accuracy because of the well matching of those subsets with the learning algorithms.

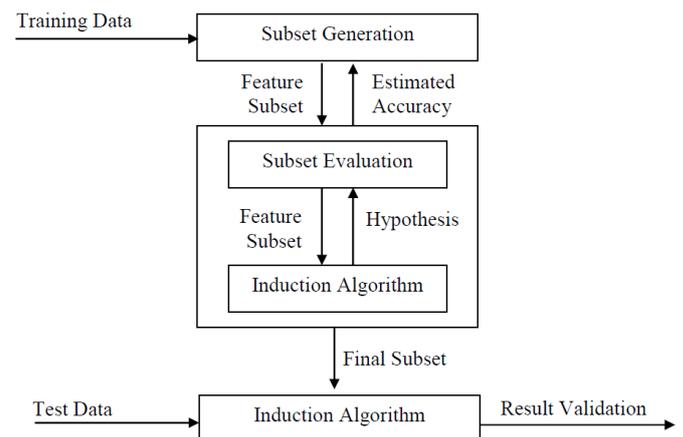


Figure 1. The wrapper approach for feature selection

The easiest method among all the wrapper selection algorithms is the forward selection (FS). This method start the procedure without having any feature in the feature subset and follows greedy approach so that it can sequentially add features until no possible single feature addition results in a higher valuation of the induction function. Backward elimination (BE) begins with the complete feature set and gradually removes features as long as the valuation does not degrade. Description about Forward Selection (FS) and Backward Selection (BS) can be found in [11] where the authors proved that wrapper selection methods are better than those methods having no selection.

Starting with an empty set of features the Best First Search (BFS) produces every possible individual feature extension [12]. BFS exploits the greedy hillclimbing approach in conjunction with backtracking to search the space of feature subsets. BFS has all the flexibility to start with an empty subset and search in forward direction. Alternatively it can start having full set of attributes and search in backward direction or it can start randomly from any point and move towards any direction. Extension of

the BFS is the Linear Forward Selection (LFS). A limited number of attributes  $k$  are taken into consideration by LFS. This method either select the top  $k$  attributes by initial ordering or carry put a ranking [13,14].

Subset Size Forward Selection (SSFS) is the extension of LFS. SSFS carries out an internal cross-validation. A LFS is executed on every fold to find out the best possible subset-size [14,15]. Through the individual evaluations attributes are ranked by the Ranker search. It uses in combination with attribute evaluators [15].

GA performs a search using the simple genetic algorithm described in Goldberg [16]. Genetic algorithms are random search techniques based on the principles of natural selection [16]. They utilize a population of competing solutions evolved to an optimal solution. Nonetheless, GAs naturally involves a huge quantity of evaluations to get to a least. Other than all these conventional methods an unconventional approach has been experimentally verified. In this method we calculated the correlation coefficient for each (except the target attribute) of the competing attribute with respect to the target attribute of the used dataset. For this purpose, the Pearson's correlation coefficient formula is used which is described in the next section. After the attribute wise calculation, those attributes were selected as feature subset whose correlation coefficient values are positive only. The attributes having negative correlation coefficient are ignored for this case. This method was named '*Positive Correlation Coefficient Selection (PCCS)*'.

### 3. Hand Tuned and Automatic Parameter Optimization

Hand-tuning is possibly the most commonly used process for machine learning technique to attain a parameter set that generates good learning behaviour. It is all about to manually change one or a few learning parameters at a time, guided by trial-and-error and the professional prior knowledge of that machine learning technique, until the model's behaviour is acceptably close to the expected or target behaviour - or until the expert loses patience.

The vision of machine learning is building the automatic specifications from data without involving monotonous and time consuming human participation. From the knowledge of repetitive experiments in the field of machine learning illustrate that machine learning techniques require appropriate learning parameter selection for their adaptation to the particular training data sets [17]. Ample of research in machine learning has given attention on the development of automatic parameter tuning algorithms (i.e. predictive modelling) for which many algorithms and datasets having hundreds or thousands of variables [6]. Predictive modelling is the process by which a model is created or chosen to get the most accurate prediction of an outcome. Many approaches or techniques have been developed using predictive modelling program to handle the automatic parameter optimization issue and applying those in particular situations [18]. In the subsequent sections, automatic MLP and SVM machine learning tuning process is performed by DTREG (pronounced as D-T-Reg) which is a very recent and advanced predictive modelling software

regarding this issue [19]. For LMS, only manual optimization/tuning are done varying the random seed  $G$ ; sample size  $S$  is not varied or tuned because of the possibility of data inconsistency. Auto optimization or tuning is generally not practiced for LMS.

## 4. Real Life Data Collection and Analysis

To perform the experiments, data is collected from the Australian Bureau of Meteorology (BOM), the National Aeronautics and Space Administration (NASA), the National Oceanic and Atmospheric Administration (NOAA). Free data is available from National Renewable Energy Laboratory (NREL) and NASA. These are excellent for multi-year averages but perform poorly for hourly and daily basis solar radiation prediction. After analyzing the collected raw data from different sources, the data provided by the Australian largest and most diverse scientific institutions the 'Commonwealth Scientific and Industrial Research Organization (CSIRO)' were selected for the experiments to develop solar radiation prediction method. The hourly raw data have been collected for a period of 2005 to 2010. The attributes in the dataset are: average air temperature, average wind speed, current wind direction, average relative humidity, total rainfall, wind speed, wind direction, maximum peak wind gust, current evaporation, average absolute barometer, and average solar radiation. The number of features used for this research is the highest in comparison to other prediction approaches for solar power prediction found in the literature review. To estimate model accuracy precisely, the wide-ranging practice is to perform some sort of cross-validation method as well as training and testing method for error estimation. For this paper both the 10 folds cross-validation method and training (70%) and testing (30%) method are examined with the used data set. Table 1 represents the statistical properties of the CSIRO raw data.

**Table 1. Statistical description of the raw data set**

	Min.	Max.	Mean	Std. Dev
Avg. Air Temp. (DegC)	-5.8	40.1	20.47	6.99
Avg. Wind Speed (Km/h)	0	27.1	6.99	4.78
Current Wind Dir. (Deg)	0	359	158.91	103.66
Avg. Relative Humidity (%)	0	100	55.11	24.26
Total Rainfall (mm)	0	30.4	0.07	0.69
Wind Speed (Km/h)	0	24.83	5.77	4.38
Wind Direction (Deg)	0	360	169.91	109.84
Max. Peak Wind Gust (Km/h)	0	106	20.45	11.33
Current Evaporation (mm)	-1.36	1.36	0.31	0.28
Avg. Abs. Barometer (hPa)	921	1020	966.59	12.09
Solar Radiation (W/m <sup>2</sup> )	1	1660	300.75	325.17

## 5. Applying Feature Selection Techniques on the Data Set

All the research works related to solar radiation prediction select the input features or attributes randomly. Unlike the conventional way this paper experimented with the maximum number of features and found out the best possible combination of features for the individual learning models of the hybrid model. To perform the experiments for selecting significant feature sub sets for

individual machine learning technique the traditional BFS, LFS, SSFS, ranker search, GS and very own the PCCS selection methods are used. To carry out experiments three algorithms for machine learning technique namely: Least Median Square [20], Multilayer Perceptrons [21] and Support Vector Machine [22] are used.

To evaluate the degree of fit that is how well a regression model fits to a data set is usually obtained by correlation coefficient. Assuming the actual values as  $a_1, a_2 \dots a_n$  and the predicted values as  $p_1, p_2 \dots p_n$ , the correlation coefficient is known by the following Equation:

$$R = \frac{S_{PA}}{\sqrt{S_P S_A}} \quad (1)$$

$$\text{where, } S_{PA} = \frac{\sum_i (p_i - \bar{p})(a_i - \bar{a})}{n-1}, \quad S_P = \frac{\sum_i (p_i - \bar{p})^2}{n-1} \text{ and}$$

$$S_A = \frac{\sum_i (a_i - \bar{a})^2}{n-1}$$

To find out the correlation coefficient of the model, the full training set is partitioned into ten mutually exclusive and same-sized subsets. The performance of the subset depends on the accuracy of predicting test values. For every individual algorithm this cross validation method was run over ten times and finally the average value for 10-cross validations was calculated. In  $k-cv$ , a data set  $S_n$  is uniformly partitioned into  $k$  folds of similar size  $P = \{P_1, \dots, P_k\}$ . For the sake of clarity and without loss of generality; it is supposed that  $n$  is multiple of  $k$ . Let  $T_i = S_n / P_i$  be the complement data set of  $P_i$ . Then, the algorithm  $A(\cdot)$  induces a classifier from  $T_i$ ,  $\psi_i = A(T_i)$  and estimates its prediction error with  $P_i$ . The  $k-cv$  prediction error estimator of  $\psi = A(S_n)$  is defined as follows [23]:

$$\xi_k(S_n, P) = \frac{1}{n} \sum_{i=1}^k \sum_{(x,c) \in P_i} 1(c, \psi_i(x)) \quad (2)$$

where  $1(i, j) = 1$  iff  $i \neq j$  and zero otherwise. So, the  $k-cv$  error estimator is the average of the errors made by the classifiers  $\psi_i$  in their respective divisions  $P_i$ .

According to Hyndman in [23], the mean absolute error (MAE) and mean absolute percent error (MAPE) are used to measure the prediction performance; these evaluation metrics are also exercised for the experiments of this paper. The definitions are expressed as:

$$MAE = \frac{\sum_{i=1}^n |p_i - a_i|}{n} \quad (3)$$

$$MAPE = (\sum |PE|) / n \quad (4)$$

where  $PE = (E / a) * 100$

$E = (a - p)$

$a = \text{Actual values}$

$p = \text{Predicted values}$

$n = \text{Number of occurrences}$

Error of the experimental results was also analyzed according to mean absolute scaled error (MASE) [24].

MASE is scale free, less sensitive to outlier; its interpretation is very easy in comparison to other methods and less variable to small samples. MASE is suitable for uncertain demand series as it never produces infinite or undefined results. It indicates that the prediction with the smallest MASE would be counted the most accurate among all other alternatives [24]. Equation 5 states the formula to calculate MASE.

$$MASE = \frac{MAE}{(1/n-1) \sum_{i=2}^n |a_i - a_{i-1}|} \quad (5)$$

$$\text{where, } MAE = \frac{1}{n} \sum_{i=1}^n |p_i - a_i| \quad (6)$$

### 5.1. Prediction of the Machine Learning Techniques using the Selected Feature Subsets

Various feature sub sets were generated or selected using different wrapper feature selection methods. Afterwards six hours ahead solar radiation prediction by the selected machine learning techniques namely LMS, MLP and SVM were performed. For this instance the selected feature sub sets were supplied to the individual machine learning techniques. The intention of this experiment was to observe whether this initiative produces any improvement in the error reduction of those selected machine learning techniques or not. For these experiments the tuning any of the particular algorithms to a definite data set was avoided. For all the experiments default values of learning parameters were used. In general, in the following tables, one can see the  $CC$ ,  $MAE$ ,  $MAPE$  and  $MASE$  of six hours in advance prediction for each machine learning technique supplied with different feature subsets. For all the experiments “W” is used to indicate that a particular machine learning technique supplied with the selected feature subsets statistically outplays the one without applying feature selection (WAFS) methods. Table 2 and Table 3 represent the obtained  $CC$  and  $MAE$  for applying LMS, MLP and SVM machine learning technique for six hours in advance prediction on the used data set before and after feature selection process.

In Table 4 and Table 5, the  $MAPE$  and  $MASE$  are shown before and after feature selection processes are applied to LMS, MLP and SVM machine learning technique for the same purpose.

The results from the experimental results show that the PCCS is somewhat superior feature selection method for LMS algorithm considering all the instances. It is noticeable that all the feature selection methods contributed to improve the  $CC$  of LMS algorithm. However, in the case of  $MAE$  all the selection algorithms except the GS improve the results for LMS. In both the case of  $MAPE$  and  $MASE$ , BFS is the only selection method which does not improve the results for LMS. It is found from those results that the Ranker Search is to some extent superior feature selection method for MLP algorithm. It is noticeable that all the feature selection methods present nearly close  $CC$  for MLP algorithm but in the case of  $MAE$ ,  $MAPE$  and  $MASE$  Ranker search is the only selection method which improves the results. Finally the obtained results illustrate that again the Ranker

Search is to some extent superior feature selection method for SVM. It is also noticeable that all the feature selection methods present either nearly close or equal *CC* for SVM.

However, in the case of *MAE*, *MAPE* and *MASE*, LFS is the only one which is unable to improve the results for SVM.

**Table 2. Achieved *CC* after applying various wrapper selection methods on LMS, MLP and SVM**

	WAFS	BFS	LFS	SSFS	Ranker		GS	PCCS	
LMS	0.95	0.96	0.96	0.96	0.96		0.96	0.97	W
MLP	0.98	0.97	0.97	0.98	0.99	W	0.97	0.98	
SVM	0.96	0.96	0.96	0.96	0.97	W	0.96	0.96	

**Table 3. Achieved *MAE* after applying various wrapper selection methods on LMS, MLP and SVM**

	WAFS	BFS	LFS	SSFS	Ranker		GS	PCCS	
LMS	77.19	76.81	74.49	74.12	74.93		87.53	63.37	W
MLP	91.02	168.34	222.73	119.11	74.31	W	288.83	110.57	
SVM	126.88	123.46	129.51	123.42	102.12	W	125.59	124.52	

**Table 4. Achieved *MAPE* after applying various wrapper selection methods on LMS, MLP and SVM**

	WAFS	BFS	LFS	SSFS	Ranker		GS	PCCS	
LMS	17.65	19.53	17.08	17.04	16.93		17.49	16.82	W
MLP	20.17	50.53	41.83	23.46	17.87	W	32.83	21.5	
SVM	21.72	21.53	22.35	21.35	20.88	W	21.35	21.65	

**Table 5. Achieved *MASE* after applying various wrapper selection methods on LMS, MLP and SVM**

	WAFS	BFS	LFS	SSFS	Ranker		GS	PCCS	
LMS	0.63	0.71	0.61	0.6	0.61		0.62	0.49	W
MLP	0.74	2.35	1.81	0.97	0.58	W	1.37	0.9	
SVM	1.03	1.02	1.05	1	0.88	W	1	1.01	

## 5.2. Prediction Results: Before versus After Applying the Feature Selection Techniques

In **Table 6** the prediction errors (*MAE* and *MASE*) of the individual machine learning techniques are compared on the basis of without supplying selected feature subsets and after supplying selected feature subsets on them. The comparative results show that errors are reduced for all the instances after supplying selected feature subsets. The terms *MAE\_BEFORE* and *MASE\_BEFORE* represent the results for without having any selected feature subsets for *MAE* and *MASE* respectively where as the terms *MAE\_AFTER* and *MASE\_AFTER* represent the results having selected feature subsets for *MAE* and *MASE* respectively.

**Table 6. Error measurements of the top most three decisive regression algorithms' prediction accuracy with feature selection**

	MAE_BEFORE	MAE_AFTER	MASE_BEFORE	MASE_AFTER	RANK
LMS	77.19	63.37	0.63	0.49	1
MLP	91.02	74.31	0.74	0.58	2
SVM	126.88	102.11	1.03	0.88	3

The subsequent sections handle the second phase of the experiments that is optimizing machine learning parameters and the prediction accuracy of those machine learning techniques are re-evaluated through adopting both the optimized parameter settings and selected feature subsets. The results obtained from the two phases of experiments are also compared in the subsequent sections to observe the gradual improvement of the accuracy of the machine learning techniques for solar power prediction.

## 6. Optimizing Different Parameters of the Selected Machine Learning Techniques

As discussed earlier, in this section the most important and influential learning parameters of MLP and SVM will be automatically tuned or optimized using *DTREG* and the only changeable parameter of LMS will be hand tuned. To the best of knowledge this is the first time that *DTREG* is used for systematic parameter tuning for MLP and SVM.

### 6.1. Parameter Optimization of MLP

Parameter structuring or parameterisation is one the classical problems for MLP machine learning technique. The key limitation of the multilayer perceptron regression model is its degrees of freedom in parameterisation. This means selecting or finding right quantity of levels, quantity of neurons within every level, learning rate, momentum constant, initial weights, activation function and bias value. Solution may not converge unless parameters are suitably selected. The MLP network is trained to search for a set of weights. These weights will be helpful to have outputs from MLP which will be very close to the actual output. A number of issues need to be considered for designing and training [25] MLP networks which are described below in correspondence with this paper and experiments:

Selection of the number of hidden layers required.

Decision of the number of neurons to be used in each hidden layer.

Searching for a globally optimal solution that bypasses local minima.

Convergence towards an optimal solution in a sensible period of time.

#### 6.1.1 Selection of the Number of Hidden Layers

In general, one hidden layer is enough to handle almost all sorts of problems. The utilization of two hidden layers hardly contributes any improvement to model and this may create the possibility of converging into 'local minima'. Theoretically no motivations are observed in support of using more than two hidden layers [25].

Nonetheless, *DTREG* provides the option to create a model with more than one hidden layers. In this section the experiments are carried on to design and train an MLP network with three layers including one hidden layer. Table 7 displays a summary of the options and parameters those were selected on the property page of *DTREG* for the MLP model.

**Table 7. Summary of the options and parameters selected for the MLP model**

Type of model	Multilayer Perceptron Network (MLP)
Number of layers	3 (1 hidden)
Hidden layer 1 neurons	Search from 2 to 30
Hidden layer activation function	Logistic
Output layer activation function	Linear
Type of analysis	Regression
Validation method	Cross validation
Number of cross-validation folds	10

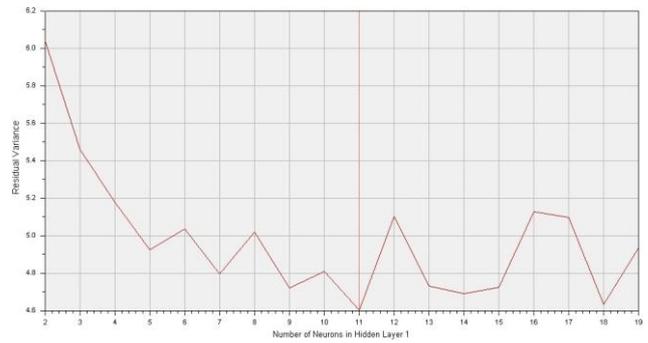
**6.1.2. Decision of the Number of Neurons to be used in Each Hidden Layer**

One of the most important aspects of MLP architecture is the number of neurons in the hidden layer. Applying the inadequate number of neurons in the hidden layer will produce incompetent and poor fitting model. On the other hand utilizing a large number of neurons may significantly increase the training time as well as over fitting model. The above mention dilemma that is discovering the moderate or optimum numbers of neurons for the hidden layer is tackled by *DTREG* with the inclusion of an automated characteristic. Table 8 demonstrates the model size summary report generated using *DTREG*. It shows that the MLP network architecture is optimal having 11 neurons for the hidden layer 1. MLP network size evaluation was performed using 4-fold cross-validation.

**Table 8. Model Size Summary Report**

Hidden layer 1 neurons	% Residual variance
2	6.04074
3	5.46065
4	5.17803
5	4.92497
6	5.03781
7	4.79645
8	5.02134
9	4.72315
10	4.80855
11	4.60414 ← Optimal size
12	5.10664
13	4.73179
14	4.69155
15	4.72458
16	5.12888
17	5.09756
18	4.63393
19	4.93682

Figure 2 graphically illustrates the error rate versus number of hidden neurons. It shows that neuron number nine in hidden layer exhibits the lowest error for MLP.



**Figure 2. Model size and error rate**

**6.1.3. Finding a Globally Optimal Solution**

The MLP architecture should be restructured in such a way that the algorithm is more likely to acquire the global solution. Even though there has been considerable research on this issue [26,27], there is no commonly established heuristic for this issue and different researchers have preference on different methodologies. A classic MLP architecture contains hundreds of initial weight values. These initial weights need to be readjusted in a way which will lead to an optimal solution. Traditional methods (e.g. steepest descent) for optimization of an MLP network are highly vulnerable to fall into the trap of ‘local minimum’. Actually those techniques are unable to observe big picture to attain ‘global minimum’. For the case of *DTREG*, the initial set of random weights is chosen by *Nguyen-Widrow* [29] algorithm. After that the optimization of those initial random weight values are performed by *DTREG* using the *conjugate gradient* algorithm which typically discovers the optimal weight values very promptly. However, the problem lays here with the fact *conjugate gradient* does not provide any guarantee of reaching to the *global minimum* [25]. To handle this condition it is really practical to bring into play *DTREG* with multiple attempts to achieve the optimization. In those manifold attempts each and every effort should have different set of random weights to start with. For the experiments of this paper the number of convergences attempts is allowed up to 4 times in *DTREG*.

**6.1.4. Convergence to the Optimal Solution**

Applying *gradient descent* algorithm on MLP actually slows down the convergence procedure and the worse is not to converge at all. Getting success on complex and large scale problems the MLP network heavily relies on the user specifications of *learning rate* and *momentum term* parameter. Automated procedure is not available to choose those parameters and wrong selection of those are responsible for extremely slow, or not any convergence at all. However, the *conjugate gradient algorithm* [25] which is used by *DTREG* to adjust the initial random weight values has been effectively applied in numerous occasions of machine learning problems and is judged one of the most efficient techniques so far conceived. The working procedure of the *conjugate gradient* algorithm usually follows a straighter pathway in comparison to the

*gradient descent* to search the optimal set of weights. Typically, the *conjugate gradient* performs considerably more robust and faster way than the *gradient descent* algorithm. One of the most important characteristics of the *conjugate gradient* algorithms is its non dependence from the user specifications of *learning rate* and *momentum term* parameters. In addition to the *conjugate gradient*, *DTREG* incorporates a latest algorithm named *scaled conjugate gradient* which was developed by MF Moller in 1993 [29]. The later algorithm exploits the numerical approximation and circumvents the unsteadiness through the integration of the power of conjugate gradient algorithm with the *model trust region* technique from *Leavenberg-Marquardt* algorithm. This combination permits the *scaled conjugate gradient* to gain reduced computational expense (i.e. it avoids the computationally expensive *line search* exercised by the conventional *conjugate gradient* algorithm) for calculating the optimal step-size towards the search direction. The experiments carried on by Moller showed that the converging speed of the *scaled conjugate* algorithm is faster twice and 20 times than the typical *conjugate gradient* and *gradient descent* algorithm respectively. Those experiments also illustrated that the frequency of failure to converge by the *conjugate gradient* is far less than the typical *conjugate gradient* and *gradient descent* algorithm.

## 6.2. Parameter Optimization of SVM

The kernel conveys earlier knowledge about the fact being modeled, determined as a similarity measure among two vectors in input space. The mapping of input features to a broader or hyperspace is done by kernel function [30] and research work is still continuing to reveal the way of selecting best possible kernel for a specific situation [31]. Even though Support Vector Machines require only very few user specified input parameters, the accuracy of an SVM model is largely dependent on the optimal selection of those input parameters and their combination (i.e. the kernel parameters such as *C*, *Gamma* and *P*).

Two very well known and established methods, *pattern search* and *grid search* are built-in *DTREG* to search the optimized learning parameters [25]. The *grid search* [32,33] works along all the values with in a prescribed search range of parameters. The *pattern search* which is also known as '*compass search*' or '*line search*' begins the searching process from the middle of the specified search area and continues test steps in all possible directions for each and every parameter value. Usually the *pattern search* involves far less assessments than the *grid search* of the model. However, the underlying limitation of a *pattern search* is the possibility of falling into *local minimum* rather than *global minimum* for the network parameters. Table 9 displays a summary of the options and parameters those were available on the property page for the SVM model.

For experiments, *Epsilon-SVR* with *Polynomial* and *RBF* kernel and *Nu-SVR* with *Polynomial* and *RBF* kernel combinations were performed with both the *Grid* and *Pattern* search to achieve optimal parameter values. Finally the *Epsilon-SVR* with *Polynomial* kernel combination was successful to provide a set of optimized parameters for SVM. Table 10 shows the optimized parameters obtained from the experiments.

**Table 9. Summary of the options and parameters available for the SVM model**

Type of model	Support Vector Machine (SVM)
Type of model	Epsilon-SVR, Nu-SVR, C-SVC, nu-SVC
SVM kernel function	Polynomial, RBF, Linear, Sigmoid
Type of analysis	Regression, Classification
Validation method	Cross validation
Number of cross-validation folds	10
Search method	Grid and Pattern
Search criterion	Minimize total error

**Table 10. Summary of the optimized parameters for the SVM model**

Optimized parameter values for SVM	
SVM model	Epsilon-SVR
Kernel function	Polynomial
Polynomial degree	3
Epsilon (P)	0.001
Cost (C)	2707.27408
Gamma (G)	0.001
Coef0	35.9381366

## 6.3. Parameter Optimization of LMS

LMS regression which is based on least squared functions is produced as of arbitrary sub-samples within raw data. For the training data, fixing the seed to select arbitrary sub-samples and setting the volume of the arbitrary examples utilized to produce the least squared functions are the major problems of LMS regression algorithm. As mentioned earlier, for LMS, only manual optimization/tuning are done varying the random seed *G*; sample size *S* is not varied or tuned because of the possibility of data inconsistency. Auto optimization or tuning is generally not practiced for LMS. Experiments were carried on with the default value of *S* = 4 and varying the value of *G* from 0 to 9. The combination *S* = 4, *G* = 6 was found to be the potential one to improve in the prediction accuracy of LMS which is demonstrated in the next section.

## 7. Prediction of the MLP, SVM and LMS with Selected Feature Subsets and Optimized Parameters

As discussed earlier, the empirical results achieved from the previous chapter demonstrate that LMS, MLP and SVM supplied with selected feature subsets provided better prediction accuracy than without having selected feature subsets. It is mentionable that for those experiments the LMS, MLP and SVM were applied with their respective default learning parameter settings. At this stage the optimized learning parameters for LMS, MLP and SVM are achieved either by hand-tuning or automatically by predictive modelling program. In this section the six hours in advance solar power prediction is again performed with the intention to achieve better prediction accuracy of the LMS, MLP and SVM by adopting both the optimized or tuned learning parameter settings and selected feature subsets on them. It is be the second level of improvement after attaining significant progress with the feature selection procedure. This particular step will help to increase the possibility of getting optimized result from the final layer of hybrid

prediction. In Tables 11 - 13 the prediction errors (*MAE*, *MAPE* and *MASE*) of the LMS, MLP and SVM respectively are demonstrated on the basis of having ‘auto optimized parameters + selected feature subsets’ combination and ‘default parameters + selected feature

subsets’ combination on them. The comparative results show that errors are significantly reduced for all the instances for the combined effect of auto optimized parameters and selected feature subsets.

**Table 11. Achieved prediction error in terms of MAE, MAPE and MASE with and without having optimized parameters for MLP**

Multilayer Perceptron Regression (MLP)					
MAE		MAPE		MASE	
Auto Optimized Parameters + Selected Feature Subsets	Default Parameters + Selected Feature Subsets	Auto Optimized Parameters + Selected Feature Subsets	Default Parameters + Selected Feature Subsets	Auto Optimized Parameters + Selected Feature Subsets	Default Parameters + Selected Feature Subsets
15.42	74.31	3.84	14.87	0.26	0.58

**Table 12. Achieved prediction error in terms of MAE, MAPE and MASE with and without having optimized parameters for SVM**

Support Vector Machine (SVM)					
MAE		MAPE		MASE	
Auto Optimized Parameters + Selected Feature Subsets	Default Parameters + Selected Feature Subsets	Auto Optimized Parameters + Selected Feature Subsets	Default Parameters + Selected Feature Subsets	Auto Optimized Parameters + Selected Feature Subsets	Default Parameters + Selected Feature Subsets
19.81	102.12	4.35	17.88	0.34	0.88

**Table 13. Achieved prediction error in terms of MAE, MAPE and MASE with and without having optimized parameters for LMS**

Least Median Square (LMS)					
MAE		MAPE		MASE	
Hand-tuned Optimized Parameters + Selected Feature Subsets	Default Parameters + Selected Feature Subsets	Hand-tuned Optimized Parameters + Selected Feature Subsets	Default Parameters + Selected Feature Subsets	Hand-tuned Optimized Parameters + Selected Feature Subsets	Default Parameters + Selected Feature Subsets
13.37	63.37	3.49	13.82	0.19	0.49

## 7.1. Synopsis of All the Experiments

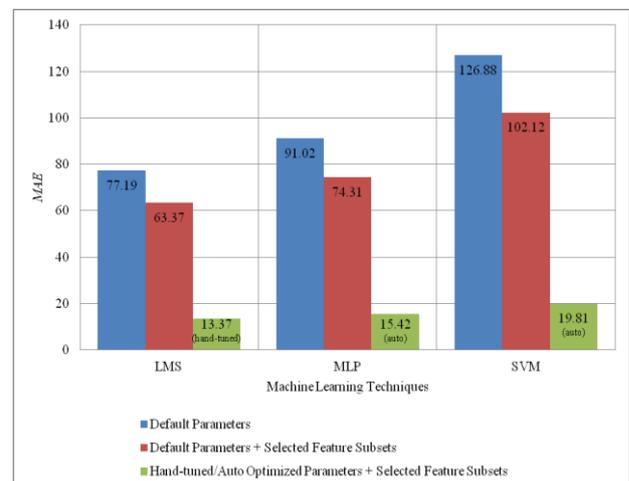
This section summarizes all the results obtained so far. The individual prediction performances of the LMS, MLP and SVM are presented in terms of *MAE* and *MASE*. As discussed in earlier sections *MAE* is considered as the standard error measurement or validation technique in this paper. In Table 14 the *MAE* of LMS, MLP and SVM are presented in three phases. In the first phase the results are shown with only the default learning parameters settings of those machine learning techniques. The second phase shows the results of each machine learning technique with optimized feature subsets and default learning parameter settings. This can be termed as ‘Optimization Level - 1’. The third phase corresponds to the results having the optimized feature subsets in combination with the hand-tuned/auto optimized parameter settings of the LMS, MLP and SVM. Again this can be termed as ‘Optimization Level - 2’. Table 15 demonstrates the results with all the above mentioned phases but those results are justified in terms of *MASE* instead of *MAE*. Figures 3 and 4 display the graphical illustration of the empirical results obtained from Table 14 - 15 respectively.

**Table 14. Summary of the prediction performances in terms of MAE**

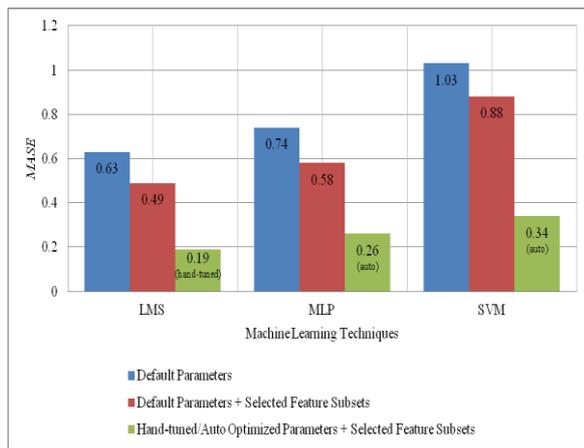
MAE			
	Default Parameters	Default Parameters + Selected Feature Subsets	Hand-tuned/Auto Optimized Parameters + Selected Feature Subsets
LMS	77.19	63.37	13.37 (hand-tuned)
MLP	91.02	74.31	15.42 (auto)
SVM	126.88	102.12	19.81 (auto)

**Table 15. Summary of the prediction performances in terms of MASE**

MASE			
	Default Parameters	Default Parameters + Selected Feature Subsets	Hand-tuned/Auto Optimized Parameters + Selected Feature Subsets
LMS	0.63	0.49	0.19 (hand-tuned)
MLP	0.74	0.58	0.26 (auto)
SVM	1.03	0.88	0.34 (auto)



**Figure 3.** Graphical illustration of the summarized prediction, performances in terms of MAE



**Figure 4.** Graphical illustration of the summarized prediction, performances in terms of MASE

## 8. Conclusions

Feature selection is a fundamental issue in both the regression and classification problem specially for the data set having very high volume of data. Applying feature selection methods on machine learning techniques may significantly contribute to increase performance in terms of accuracy. In this paper various methods of feature selection methods have been briefly described. In particular the wrappers are found better selection method which is also justified by the results obtained from the experiments performed in this paper. The results from the experiments demonstrate that LMS, MLP and SVM supplied with selected feature subsets provide better prediction accuracy (i.e. reduced MAE and MASE) than without having selected feature subsets. It is mentionable that for these experiments the machine learning techniques were applied with the default learning parameter settings. Therefore the later part of this paper continued with the extended experiments with the intention to achieve better prediction accuracy of the selected machine learning techniques by adopting both the optimized or tuned learning parameter settings and selected feature subsets on them. The new results obtained from the later stage were compared with the earlier prediction results of the same machine learning techniques those used the selected feature subsets only but not the optimized learning parameters. Empirical results suggest that the applying optimized parameters with the selected feature subsets yield excellent generalization performance of LMS, MLP and SVM in terms of MAE and MASE. In general it can be concluded that providing more attention and effort towards the potential feature subsets selection and machine learning parameter optimization (e.g. combined effect of selected feature subsets and optimized parameters on prediction accuracy which is investigated in this paper) can significantly contribute to the improvement of the accuracy for solar power prediction.

## References

- [1] Yu, L., Liu, H. Feature selection for high-dimensional data: a fast correlation based filter solution. Proc. 20th Int'l Conf. Machine Learning, 2003; 856-863.
- [2] Blum, A., Langley, P. Selection of relevant features and examples in machine learning. Artificial Intelligence, 1997; 97:245-271.
- [3] Mitchell, T. Machine Learning. McGraw Hill, 1997.
- [4] Karagiannopoulos, M., Anyfantis, D., Kotsiantis, S. B., Pintelas, P. E. Feature selection for regression problems. The 8th Hellenic European Research on Computer Mathematics & its Applications, HERCMA 2007, 20-22.
- [5] Langley, P. Selection of relevant features in machine learning. In: Proceedings of the AAAI Fall Symposium on Relevance, 1994; 1-5.
- [6] Automatic parameters selection in machine learning. Editorial / Neurocomputing, Elsevier, 2012; 75:1-2.
- [7] Kalogirou, S. a. Artificial neural networks in renewable energy systems applications: a review. Renewable and Sustainable Energy Reviews, 2001; 4:373-401.
- [8] Daelemans, W., Hoste, V., Meulder F., Naudts, B. Combined optimization of feature selection and algorithm parameters in machine learning of language. CNTS Language Technology Group, University of Antwerp.
- [9] Konen, W., Koch, P., Flasch, O., Bartz-Beielstein, T. Parameter-tuned data mining: a general framework. University of Applied Sciences, Cologne.
- [10] Tan, Feng. Improving feature selection techniques for machine learning. Computer Science Dissertations. Paper 27, 2007.
- [11] Caruana, R., Freitag, D. Greedy attribute selection. Machine Learning: Proceedings of the Eleventh International Conference, San Francisco, CA, 1994.
- [12] Kohavi, R., John, G. H. Wrappers for feature subset selection. Artificial Intelligence, 1997; 97(1-2):273-324.
- [13] Guetlein, M., Frank, E., Hall, M., Karwath, A. Large scale attribute selection using wrappers. Proc IEEE Symposium on Computational Intelligence and Data Mining, 2009; 332-339.
- [14] Guetlein, M. Large scale attribute selection using wrappers. Germany, 2006.
- [15] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, H. The WEKA data mining software: an update. SIGKDD Explorations, 2009; 11.
- [16] Goldberg, E. Genetic algorithms in search, optimization and machine learning. Addison-Wesley, 1989.
- [17] Postema, M., Menzies, T., Wu, X. A decision support tool for tuning parameters in a machine learning algorithm. The Joint Pacific Asia Conference on Expert Systems/Singapore International Conference on Intelligent Systems. (PACES/SPICIS 97) 1997.
- [18] Geisser, Seymour. Predictive inference: an introduction. New York: Chapman & Hall, 1993.
- [19] Sherrod, P. H. DTREG: Predictive modeling software.
- [20] Rousseeuw, P.J. Least median of squares regression. J. Amer. Statist. Assoc., 1984; 79:871-880.
- [21] Haykin, S. Neural networks: a comprehensive foundation. Prentice Hall, 1999.
- [22] Shevade, S., Keerthi, S., Bhattacharyya, C., Murthy, K. Improvements to the SMO algorithm for SVM regression. IEEE Transaction on Neural Networks, 2000; 5:1183-88.
- [23] Zheng, H. Y., Kusiak, A. Prediction of wind farm power ramp rates: a data-mining approach. ASME J. Solar energy Eng., 2009.
- [24] Hyndman, R. J., Koehler, A. B. Another look at measures of forecast accuracy. Monash Econometrics and Business Statistics Working Papers, 2005.
- [25] DTREG manual in PDF format.
- [26] Miller, G.F., Todd, P.M., Hedge, S.U. Designing neural networks using genetic algorithms. Proc. 3rd International Conference on Genetic Algorithms, 1989.
- [27] Rumelhart, D. E., Hinton, G. E., Williams, R. J. Learning representations by back propagating errors. Nature, 1986, 323(9):533-536.
- [28] Nguyen, Derrick, Widrow, B. Improving the learning speed of 2-layer neural networks by choosing initial values of adaptive weights. In Proc. IJCNN, 1990; 3: 21-26.
- [29] Moller, Fodslette, M. A scaled conjugate gradient algorithm for fast supervised learning. Pergamon press. 1993.
- [30] Zhang, J., Lee, R., Wang, Y. J. Support vector machine classifications for microarray expression data set. Proceedings of the Fifth International Conference on Computational Intelligence and Multimedia Applications (ICCIMA'03), IEEE, 2003.

- [31] Burges, C. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, Kluwer Academic Publishers, 1998.
- [32] Wang, J., Wu X., Zhang, C. Support vector machines based on k-means clustering for real-time business intelligence systems, *Int. J. Business Intell. Data Mining*, 2005,1(1): 54-64.
- [33] Hsu, C. W., Chang, C. C., Lin, C. J. A practical guide to support vector classification. Technical Report, University of National Taiwan, Department of Computer Science and Information Engineering, 2003: 1-12.