A neural network version of the measure correlate predict algorithm for estimating wind energy yield

J. F. Dale Addison, Andrew Hunter, Jeremy Bass, Matt Rebbeck

Centre for Adaptive Systems, School of Computing Engineering and Technology, University of Sunderland, St Peters Campus, St Peters Way, Sunderland, SR6 0DD, UK

Renewable Energy Systems Ltd., <address>

E-mail cs0dad@isis.sunderland.ac.uk

Abstract

We have investigated the feasibility of using neural networks to make predictions of long term energy yield at a potential wind farm site. This paper considers the effectiveness of neural networks in predicting wind speed at a target site from wind speed and direction measurements at a reference site. The technique is compared with the standard Measure Correlate Predict (MCP) algorithm used in the wind energy industry. Improvements of predictive accuracy in the region of 5%-12% can be achieved. Best results are obtained using multilayer perceptron networks with a large number of hidden units, with extensive Quasi-Newton (BFGS) training. Experiments have been conducted using contemporaneous measurements, and time shifted wind speed (previous and next hour) as inputs. Performance is consistently improved by using time-shifted inputs. However, the improvement in performance has to be offset against the financial penalty incurred in purchasing time series data for input.

Introduction

In recent years, naturally occurring renewable sources of energy (wind, tidal, geo-thermal) have been increasingly mooted as supplements or alternatives to fossil fuels. However because such energy sources are dependant upon a complex interaction of geographical and physical factors, locating an area which can generate a sufficiently high long term energy yield to be commercially viable is an essential requirement. In the wind energy field, a prime consideration is to find a wind farm site with a consistently high long-term wind speed. The most popular technique used for wind speed determination is the "Measure Correlate Predict" (MCP) algorithm.

[Matt, Jeremy - could you insert a brief overview of alternative techniques here, and maybe also update the stuff on MCP here if you think I have misrepresented anything - Thanks, Andrew]

In MCP [ref], a test mast is erected at the target site, and wind speed and direction are measured over a short period of time (six months to a year). Since this period is too short to make reliable decisions, the measurements are correlated with contemporaneous measurements from a nearby reference site (meteorological office data). A predictive model that takes as input reference site measurements, and outputs estimated target site values, is built.

By applying the predictive model to historical data from the reference site, a prediction can be made of long term wind energy yield at the target site.

Typically, full time-series data is not obtained for the entire historical period, as this is financially costly. Instead, a two dimensional histogram is purchased characterizing the distribution of wind speed and direction at the reference site. The predictive model is then used to map the histogram from reference to target site by transforming the key points (corners of histogram columns) to the target site, and re-binning. Finally, expected energy yield is calculated by applying turbine power generation curves to the expected wind distribution figures.

The key issue in MCP is therefore to design the predictive model. We are primarily interested in estimating wind speed.

A simple linear model, taking reference wind speed and direction as inputs is not appropriate, as the wind direction has a strongly non-linear relationship with the wind speed at the target site. In contrast, given a certain wind direction, the relationship between wind speed at the two sites is close to linear. Consequently, the standard MCP technique

sectors the data by reference wind direction, and then applies a separate univariate linear model mapping reference wind speed to target wind speed within each sector.
The overall MCP process is summarized in figure 1 below:

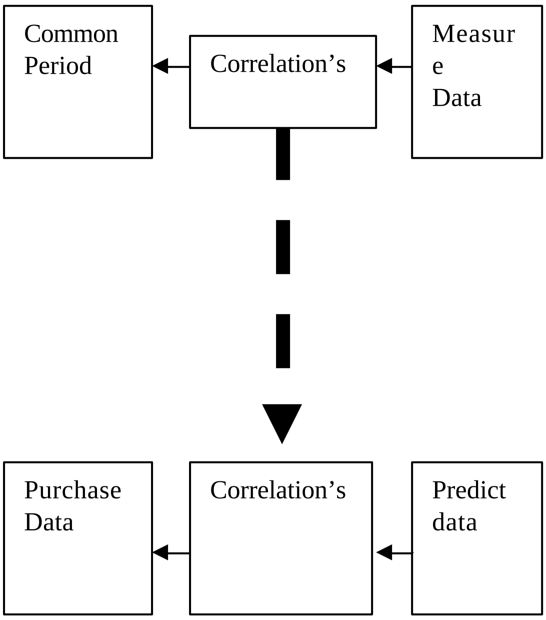

Figure 1: - Diagrammatic representation of MCP technique

Disadvantages of this technique include: the reduced volume of data available to optimise each linear model (on average one twelfth of the total data, but some sectors may have much less); and the discretization of a continuous variable which may lead to inferior performance (i.e. when the wind direction is close to the boundary of two sectors, performance may suffer by arbitrarily assigning to one sector).

Overview of Measure Correlate Predict techniques

(We need an literature overview (v. brief) of previous MCP techniques. I would suggest that Jeremy or Matt provide this.)

Architectures used

We have tested several different neural network architectures on this problem, specifically:

? Radial basis function networks [6].
? Generalised regression networks [7]
? Probabilistic neural networks [8]
? Multi layer perceptrons [9]

Results of experiments

Our experiments established very quickly that all architectures used other than multilayer perceptrons, performed poorly compared to the measure correlate predict algorithm. Consequently only the results from the Multilayer perceptron experiments are presented in this paper.
We implemented a sectored linear model as a benchmark, and then experimented with a range of neural networks with different sets of input variables. The input variables used were:
Wind direction and speed;
Wind direction, and speed at *t*?1, *t* and *t*+1.
Wind direction, single input speed smoothed over the range *t*-1, *t*, *t*+1.
Two input trigonometric representation, *s*.sin(?), *s*.cos(?).

The major experiments, reported below, used the two input, and four input with lagged wind speed models. Experiments indicated that a substantial number of neurons are needed in the hidden layer, and consequently that a long period of optimisation is required. We increased the number of hidden layer neurons iteratively in steps of 30, stopping once error levels deteriorated. Network performance was assessed by the Pearson-R correlation between the predicted and expected target site wind speeds, using an independent test set not deployed in training. We experimented with a variety of training algorithms. Optimum performance was achieved via the use a combination of a short burst of back-propagation [11] (20 epochs) followed by a large number of iterations (2000-3000) of Quasi-Newton BFGS [12]. Alternative algorithms such as Conjugate Gradient Descent did not have acceptable performance.
Experiments were conducted on seven data sets, reflecting different wind conditions in a variety of European locations. The results are summarised in table 1.

Input Units	Hidden Units	NN Corr.	MCP Corr.
4	60	76%	74%
4	60	78%	-
4	300	82%	80%
2	30	18%	15%
4	80	86%	86%
4	150	98%	96%
4	150	99%	97%

Table 1: - Best neural networks, showing number of inputs and hidden neurons

In all cases, the neural network approach works at least as well as MCP, and in some cases considerably better. This seems at first a little surprising, as the data for a particular wind direction sector appears very close to linearly distributed, until one realises that the non-linear relationship with wind direction is exploited by the neural network, which models a continuous surface, whereas the MCP technique quantizes wind direction and is therefore more prone to make errors at directions close to the boundaries between sectors.

Another key finding was that including lagged variables usually improves performance, with the four input models performing best on six of the seven data sets. One possible explanation for this could be that the wind speed is noisy, and some smoothing is therefore helpful. To test this theory, we also evaluated a two input model using a time-smoothed reference speed as input. This model performed no better than the standard two input version. We therefore conclude that the inclusion of lagged variables conveys some additional information. X [ref] has shown that the prevailing weather pattern has a key effect on the relationship between the variables, and it is possible that lagged wind speed variables convey some information about the current weather pattern, as the time-evolution of wind speed is determined by the pattern. Certainly, an examination of the correllelogram demonstrates that there is a significant time-based structure to the errors made all the models we have investigated, and this indicates that key factors are not captured within the available data.

The improvement in performance when using time lagged variables presents a difficult decision regarding deployment of the technique. To apply the model with time lagged variables for prediction of typical wind speeds throughout a historical period, we need to purchase full Time Series data for the entire period, whereas the standard two-input model can be applied to a summary histogram, and full Time Series data is substantially more expensive than a histogram. For practical application, therefore, we would need to assess carefully the expected reduction in risk (and its financial consequences) versus the cost of purchasing the data.

One apparent problem with the technique we have used is the treatment of wind direction, which enters the model as a single input variable (normalised to the range [0,1]). The extremes are close together in angle, with 0^o and 360^o equal, which suggests that angle should be treated specially. We therefore also experimented with a trigonometric reformulation of the input variables, where the components *s*.sin(?) and *s*.cos(?) are used as inputs. However, this model proved to have inferior performance to the standard two-input approach, and so was abandoned.

Of particular interest is the number of neurons required to carry out the modeling of the response surface. Wind data sets six and seven require 150 hidden neurons, whilst wind data set three required 300. Training times varied from 30 minutes up to seven hours. This depended upon the number of neurons in the middle layer, and the type of data set being processed (i.e. degree of non-linearity).

The number of neurons required for the solution of this problem is quite exceptional, and unprecedented in our experience. We consider the reason for this is the degree of linearity (or otherwise) in the problem domain. In some cases our data sets are quite close to linear, and counter-intuitively this appears to require a large number of neurons to model the surface, as some quite small perturbations in the surface are required. In contrast, the data sets that are more non-linear require less neurons in the hidden layer.

Summary and conclusions

We have carried out a number of experiments into the use of neural networks for long term energy prediction of wind direction and speed data collected from potential wind farm sites. These data are then used with contemporaneous data collected at a reference to optimise a predictive model, and further target data is inferred using a historical model. We have modified the standard MCP (measure correlate predict) algorithm, and have created out own version using a linear neural network, creating a neural network version. Multi-layer perceptrons trained using a combination of back-propagation (very short burst) followed by extensive Quasi-Newton training produce the best results. Time-lagged wind speed inputs further improve performance, indicating that key structural information about prevailing weather patterns is missed. Nevertheless, our experiments indicate that neural networks do offer models of improved accuracy in this problem domain.

References [HAVEN'T RECONCILED THESE AGAINST CITATIONS PROPERLY YET. PLEASE ADD ANY THAT SHOULD BE IN HERE]

[1] Tande J O G.; and Landberg L 1993, A 10 sec forecast of wind turbine output with neural networks, *European community wind energy conference,* 8-12 March : 774-777.

[2] Lloyd W.;1995, Wind resource assessment using measure correlate predict techniques, *MSc Renewable energy systems technology,* Department of Electronic and Electrical Engineering, University of technology, Loughbrough, Leicestershire; 12-13.

[3] Landberg L.; 1992 Neural networks utilised for the prediction of a geophysical time series, A comparison, Ris? -R-656(DA). Ris? National laboratory, Roskilde, Denmark,76.

[4] Satinderpaul S.; Degvan and Dusty R.; Walker Jr, 1996 Design of a hybrid wind power storage and generation system for a remote community, *Applications of Neural networks in environment energy and health,* 135-144

[5] Golub, G.; and Kahan, W. 1965. Calculating the singular values and pseudo-inverse of a matrix, *Numerical Analysis, B* 2 (2): 205-224

[6] Lowe, D.; 1995 Radial Basis Function Networks, The Handbook of Brain Theory and Neural Networks, Cambridge, MA: MIT Press.

[7] Specht, D.F. (1991) "A Generalized Regression Neural Network", IEEE Transactions on Neural Networks, 2, Nov. 1991, 568-576

[8] Specht, D. F.; 1990 Probabilistic Neural Networks. *Neural Networks* 3 (1): 109-118.

[9] Haykin, S.; 1995 Neural networks-A Comprehensive foundation. Maxwell Macmillan International, publishing Co, 138

[10] Hestenes, M. R.; and Stiefel, E. 1952 Methods of Conjugate Gradients for Solving Linear Systems. *Journal of Research of the National Bureau of Standards* 49(6): 409-436

[11] Rumelhart, D. E.; Hinton, G. E.; and Williams, R J. 1986. Learning Internal Representations by Error Propagation in Parallel Distributed Processing: Explorations in the *Microstructure of Cognition:* 318-362.

[12] Barnard, E.; 1992 Optimization for Training Neural Nets. *IEEE Transactions on Neural Networks* 3(2), 232-240

.