

# Short-Term Wind Power Prediction with Radial Basis Networks

Rutger W. ter Borg<sup>a</sup> and Léon J.M. Rothkrantz<sup>b</sup>

January 27, 2008

<sup>a</sup>Nuon Energy Sourcing, Spaklerweg 20, Amsterdam, The Netherlands

<sup>b</sup>Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands

## Abstract

In this paper, we propose a method to predict wind power production with radial basis function networks. In this case, the power production is the aggregated production of all wind farms of one electricity company. The method uses wind speed predictions supplied by a meteorological agency, and predicts up to several days ahead. The coarse resolution of one meter per second is overcome by combining the weather data from several meteorological stations. The wind direction is mapped on a circle so it is more compatible with a radial basis. These ingredients have been combined with a kernel machine, which has been implemented and tested. Test results are presented in the paper.

**Keywords:** Wind Power Prediction, Radial Basis Functions, Kernel Machines

## 1 Introduction

The endeavour to reduce the amount of carbon dioxide in the atmosphere [8] has been ongoing for a while. The European Union (EU)'s renewables directive has been in place since 2001 [9]. It aims to raise the share of electricity produced from renewable energy sources (RES) in the EU to 22% by 2010. The efficiency of wind power turbines have been significantly improved during the last decade, and have become an attractive source of renewable energy. At the moment, wind power is the fastest growing type of renewable energy in Europe.

However, wind power also has the other side of the coin: because the amount of energy produced strongly depends on the actual wind speed at a certain location, the power output cannot be guaranteed at all times. The variation in power production causes noise to be transmitted onto the electricity grid, and has to be counterbalanced by flexible and expensive power plants. With an accurate short-term forecast of wind power, an improved economic dispatch of generating units will be possible, saving fuels and the environment. We consider

hourly wind power forecasts up to 48 hours ahead. The value of the produced wind energy will be higher when the delivery has a higher guarantee.

Radial basis function networks are an advanced variant of artificial neural networks that have excellent nonlinear approximation capabilities. They have been successfully applied to a large diversity of problems, including chaotic time series modelling [5]. Kernel machines have drawn a considerable amount of attention in recent years. Perhaps because kernel machines are relatively new, they are rarely found in the wind power prediction literature. Emerged by the combination of several disciplines, kernel machines have in common that they combine the kernel trick [1] and the principle of parsimony [19, 23].

Section 2 discusses typical approaches to short-term wind forecasting, including what kind of data is used and the layout of forecasting systems. A brief introduction to radial basis networks and kernel machines is given in section 3. Section 4 describes the setup of a wind power forecasting system with kernel machines. Experimental results are shown in section 5. Section 6 concludes the paper.

## 2 Wind Power Prediction

An exhaustive literature overview of wind power prediction is available in a report by Giebel et al. [10]. With respect to short-term forecasting of wind power, we identify two different approaches: the physical approach and the statistical approach. In the physical approach, one makes a description of the dynamics of the underlying system, based on complete knowledge of all its subsystems [3]. In the statistical approach, models are constructed from the data, without specific domain knowledge.

### 2.1 The Physical Approach

Generally, in physical approach, the underlying system is decomposed in three different subsystems. First, the wind speeds at turbine height are estimated in the scale down phase. In the second step, these estimated wind speeds are converted to a power estimate, followed by a third step, where the power estimates are aggregated to reflect multiple wind farms. We will discuss these steps in more detail below.

- **To scale down.** To estimate the wind power produced at some site, ideally one wants to have wind speed measurements done on that exact location. Let us call these  $wind_i(t)$ . During the downscaling phase, numerical weather forecasts are adjusted to match the turbine hub height. Often weather data is obtained by using a transformation  $f$  on weather predicting models such as HIRLAM, i.e.,  $wind_i(t) = f(HIRLAM_i(t))$ . The error made by the HIRLAM model is not available, because no actual measurements are available.

- **To convert to power.** In this step, it is assumed that the total amount of electricity produced by the turbines depends on wind speed only. Using the so-called power-curve  $p$  of the turbines, the wind speeds are converted to an expected power production, i.e.,  $\text{power}_i(t) = p(\text{wind}_i(t))$ . Power curves are often obtained from the supplier of the wind turbines, or are estimated on the basis of measurements on the wind turbines themselves [16, 15].
- **To scale up.** The estimated power of a subset of wind turbines or farms is scaled to match the installed total wind power production capacity, and then the efficiency of the wind farm to finally determine the farm's total output. This problem can be expressed as  $\text{total power}(t) = s(\text{power}_1(t), \text{power}_2(t), \dots, \text{power}_N(t))$ , with  $s$  the scale-up function. This specific step has also been addressed by more sophisticated tools such as a fuzzy-neural network [18].

Although the results obtained are acceptably good, it requires a lot of data management of all individual farms and weather conditions.

## 2.2 The Statistical Approach

In the statistical approach, the model is inferred from the data. The common approach with this one-stage type of approach is to use neural networks [16] or another kind of regression technique [2]. It usually estimates the wind power production in one step, by taking the numerical weather predictions and transforming them to the estimated wind power production.

- **To learn.** On the basis of input vectors  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N \in \mathcal{X}$  and the corresponding observed scalars  $y_1, y_2, \dots, y_N \in \mathbb{R}$ ,  $f$  is fitted to the observed data assuming an additive error

$$y = f(\mathbf{x}, \mathbf{w}) + \varepsilon \tag{1}$$

and parametrised by some parameter vector  $\mathbf{w}$ . In the case of modelling wind power production, the inputs usually contain (actual or predicted) wind speeds, the outputs contain measured wind power production. Generalisation to unobserved measurements involves the difficulty of computational learning [21, 13], which is commonly addressed by cross-validation techniques.

This class of models work in a more implicit way: the numeric weather predictions can be translated to a wind power in one single step. A drawback of this approach is that it is a black box, i.e., its one and only function is to predict the expected wind power production.

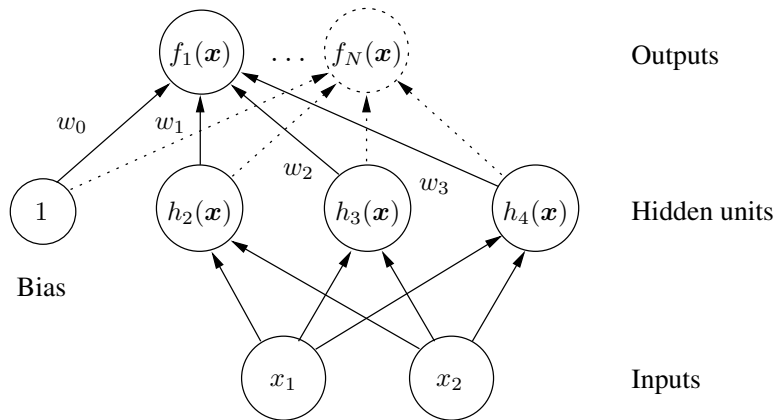


Figure 1: An RBF network with inputs in  $\mathbb{R}^2$ , a bias, and 3 hidden units, and a single output. Dashed lines show an RBF network architecture for multiple outputs.

### 3 Modelling with Radial Bases

The property of a radial basis is that the function value decreases (or increases) monotonically with the distance from some central point. We will discuss basic radial basis network terminology in subsection 3.1, and its similarity to and brief introduction to kernel machines in subsection 3.2.

#### 3.1 Radial Basis Function Networks

Radial basis function (RBF) networks have traditionally been associated with radial basis functions in a single layer network such as shown in Figure 3.1. In the input layer, each element of the input vector  $\mathbf{x}$  is fully connected to all inputs of the hidden layer neurons. RBF network topology is determined by the number of hidden units. In the hidden layer, the hidden unit activation function  $h_i(\mathbf{x})$  is a radial basis function. The output layer combines a scalar-valued bias  $w_0$  and the outputs of the functions in the hidden layer, to form

$$f(\mathbf{x}) = w_0 + \sum_{i=1}^N w_i h_i(\mathbf{x}).$$

The network parameters are established by minimising a cost function

$$\min \sum C(y, f(\mathbf{x}))$$

which is typically the sum of the squares of the residuals. As with classical artificial neural networks, RBF networks can be trained by a variety of supervised learning algorithms. In the initial approaches, all data samples were assigned to the hidden layer to act like a centroid. In later approaches, the number of

hidden units was reduced by the use of clustering algorithms such as  $k$ -median [4], or by stochastic choice [11]. Other algorithms are orthogonal least squares [20] and gradient descent [12].

### 3.2 Kernel Machines

When considering Gaussian kernel functions,

$$k(\mathbf{u}, \mathbf{v}) = \exp\left(-\frac{1}{2}\sigma^{-2}\|\mathbf{u} - \mathbf{v}\|_2^2\right), \quad (2)$$

kernel machines can be seen as a topology adaptive approach to radial basis networks, with the locations of the radial bases restricted to the set of inputs. Criteria for selection of the hidden units in this case is either structural risk minimisation of Vapnik [23], or automatic relevance determination done by Bayesian sparseness inducing priors [17].

Kernel machines combine statistical learning theory to optimise generalisation [22, 23, 24], mathematical programming to find solutions efficiently, and the *kernel trick* to handle non-linearity [1]. Variants of the support vector machine have been introduced [14], as well as variants of Bayesian sparseness inducing methods [7]. The Bayesian methods tend to produce more accurate and concise results than the support vector machine. However, they are computationally more costly.

In case of regression, kernel machines use the fact that observational data can be represented by a linear combination of *kernel functions*  $k$  [25, 6]

$$f(\mathbf{x}) = w_0 + \sum_{i=1}^N w_i k(\mathbf{x}_i, \mathbf{x}) \quad (3)$$

with  $w_i \in \mathbb{R}$  and  $\mathbf{x}_i \in \mathbb{R}^N$ . One can, but often does not have to, deliberately design the similarity of points in the state space by altering this kernel function.

Kernel machines exploit the idea of mapping data to a high-dimensional feature space where some linear algorithm is applied that works exclusively with inner products. Suppose we have some mapping  $\Phi$  from an input space  $\mathbb{R}^N$  to a feature space  $\mathcal{H}$ , then a *kernel function* (or kernel)

$$k(\mathbf{u}, \mathbf{v}) = \langle \Phi(\mathbf{u}), \Phi(\mathbf{v}) \rangle$$

is used to define the inner product in feature space  $\mathcal{H}$ . Figure 3.2 illustrates the basic idea of the kernel trick: the result of applying an inner product in feature space  $\mathcal{H}$  corresponds to a nonlinear estimate in input space  $\mathbb{R}^N$ . In our case we will consider the Gaussian kernel.

## 4 Model Setup

This section consists of two parts, first an analysis of the data at hand, and second the proposed model.

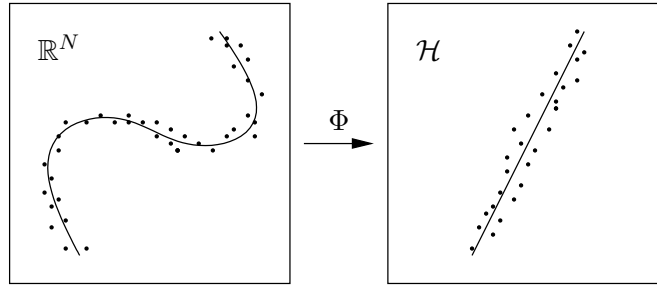


Figure 2: A mapping  $\Phi$  from input space  $\mathbb{R}^N$  to feature space  $\mathcal{H}$ .

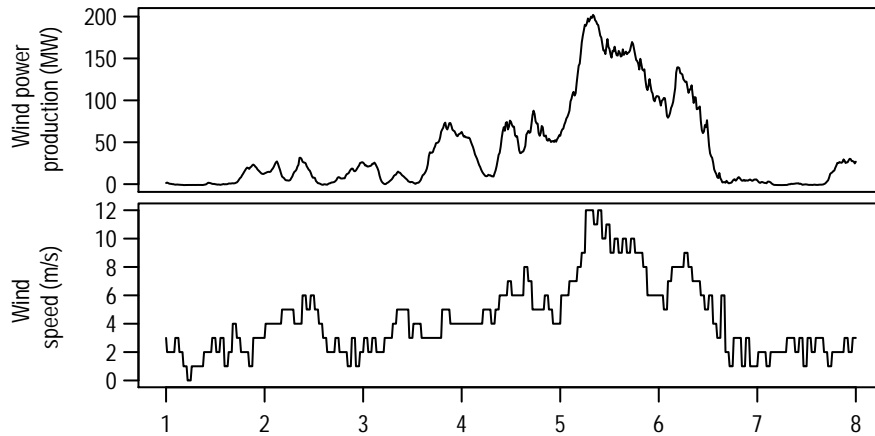


Figure 3: Seven days of wind power production (top), and their corresponding measured wind speeds at the same time (bottom).

#### 4.1 Data Analysis

A large number of telemeters measured the exact electricity that has been put on the electric grid. Measuring is done by grid operators, they supply these data to the owners of the wind farms. In data-driven modelling, as done in this paper, data are used to create models with. The acquisition and verification of the production data of individual wind-power turbines is a tremendous undertaking: it all has to be checked for meter outage, and all time series have to be verified to actually be wind-power turbines. Figure 4.1 shows the produced wind power and wind speeds for one week.

It illustrates produced amounts of energy that are typical to wind power, i.e. on the first day, almost no energy production occurred, while on the fifth day, the production is often well above 150 MW. The variation in the power produced is significant. When wind speeds are not sufficiently high, no production takes place at all. Some types of wind turbines *consume* energy in case of low wind

Table 1: Correlation matrix of available weather stations and the wind production.

	Amsterdam	De Kooy	Stavoren	Valkenburg	Production
Amsterdam	1.00	0.851	0.810	0.875	0.864
De Kooy	0.851	1.000	0.821	0.799	0.881
Stavoren	0.810	0.821	1.000	0.791	0.857
Valkenburg	0.875	0.799	0.791	1.000	0.831
Production	0.864	0.881	0.857	0.831	1.000

speeds.

We do not have any wind measured at the parks themselves, but rather measured at weather stations operated by the Dutch meteorological institute. Traditionally wind speeds were measured in knots<sup>1</sup>. The actual wind speeds obtained from a measuring device are corrected for the surrounding surface and installation height. This results in the potential wind speed. The correction factor is usually between 0.9 and 1.2. The distribution of the potential wind speed therefore will be clustered around the original value in knot. The potential wind speed is reported with an accuracy of 0.1 meter per second, but the resolution of the records from which it was computed is approximately 0.5 meter per second. From July 1996 wind speeds are measured in integer values of meter per second. So, since then the resolution is even smaller. We have available wind speeds measured from the four stations mentioned in table 1. This table also displays the correlations between the wind speeds measured at each station, and the recorded wind power production.

## 4.2 Feature Space

This lack of resolution in wind speed records as mentioned in subsection 4.1 is one of the first issues one encounters when creating a wind forecasting system. If this resolution is not improved, the predictions will suffer from the same discretion level. Figure 4 (left) illustrates this by the power curve of the data using the wind speed of the Netherlands. We propose to overcome the problem of coarse resolution by taking into account multiple weather stations. In Figure 4 (right), we illustrate this step by showing the wind-power curve of the average wind speed of all weather stations presented in table 1. Averaging is done merely for illustration purposes, the model will use multiple inputs, i.e., not an averaged number.

The direction of the wind is recorded in degrees to North. In order to optimise this entity to work in combination with a radial basis function, we will map the polar representation to an Euclidean representation during a pre-processing step. The exact setup of the feature space for the wind power prediction system will be done in an experimental way.

<sup>1</sup>Source: website KNMI <http://www.knmi.nl/>.

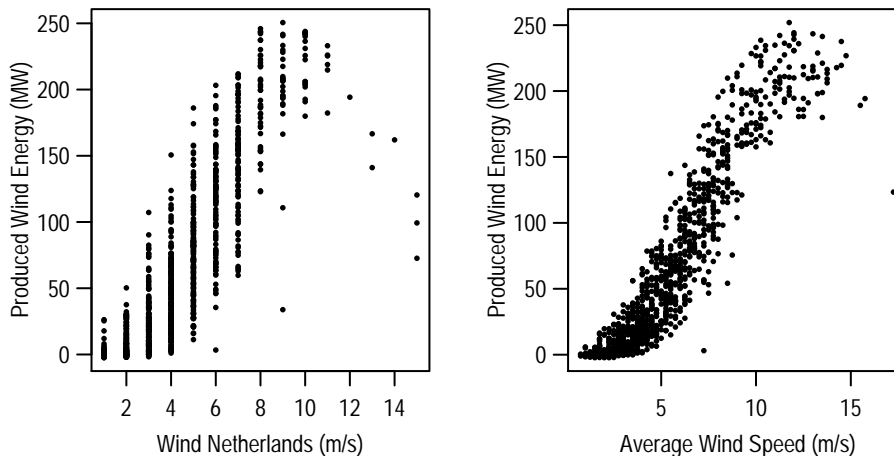


Figure 4: A wind power curve using the discretised wind speeds (left), and using the computed average wind speeds (right).

Table 2: Errors measured over the fitted models. The root-mean square error (RMSE), mean-absolute error (MAE) and maximum-absolute error (MAX) are reported in megawatts (MW). The statistical correlation (Cor) is unitless.

added station	No wind direction				With wind direction			
	RMSE	MAE	MAX	Cor	RMSE	MAE	MAX	Cor
Stavoren	30.78	22.68	174.57	0.886	28.07	20.55	191.78	0.907
De Kooy	21.83	16.08	137.41	0.945	21.33	15.74	137.53	0.947
Amsterdam	19.39	14.37	109.07	0.957	18.97	14.08	98.65	0.959
Valkenburg	18.96	14.11	100.39	0.959	18.64	13.86	93.97	0.960

## 5 Experimental Results

To select the features used by the wind power prediction system, we have conducted several experiments on the data. We have initialised the Gaussian kernel that we used with parameter  $\sigma = 1$ . The wind speeds are scaled with a factor of 0.05. The diameter of the wind direction circle is set to 0.5. We have tested with cross-validation that these settings performed reasonably well. Table 2 shows the results of fitting with different feature spaces. On the left side of the table, the results are shown without the wind direction taken into account, while on the right side of the table the wind direction is also taken into account. The error measures used were root-mean-square of the error (RMSE), the mean absolute error (MAE), the maximum absolute error (MAX) and the statistical correlation (Cor). All errors reported are expressed in megawatts (MW), except the statistical correlation, which is unitless.

Common benchmarks in determining the quality of the wind power prediction are the persistence model and the mean-production model. When using

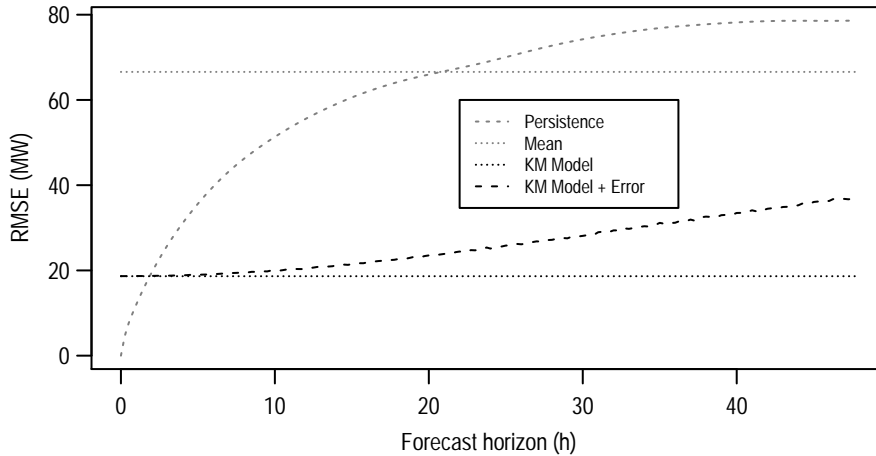


Figure 5: Root-mean-square of the error of different models.

persistence, one takes the previous measured value(s) as the prediction for the next value(s). The persistence is commonly the model to beat [10]. The mean-production model simply reproduces the mean of the production at all times. We have used the kernel-machine model using an input space of four weather stations and wind direction, of which the fitting errors are shown on the right of the bottom row of table 2.

Because we did not have available the historically predicted wind speeds and directions, we simulated an error development in the numerical weather forecasts. To do so, we have corrupted the wind speed measurements for each forecast horizon step with Gaussian multiplicative noise,  $w_c(t+i) = w_m(t+i)\mathcal{N}(1, c \cdot i)$  with  $w_m$  being the measured wind speed at time  $t$ ,  $i$  being the number of hours ahead, and with  $c$  a constant indicating the severity of error development. During experiments we have set  $c$  to  $1/120$ .

Figure 5 shows the root-mean-square of the errors of four different models: the persistence model, the mean of the production measurements, the kernel-machine model, and a kernel-machine model with corrupted wind measurement data. It illustrates that the forecast time horizon does not affect the mean-load and kernel machine model. Persistence is the best model for approximately the first two hours, after which the kernel-machine model has the lowest error. The error caused by the multiplicative noise on the wind speeds does not cause dramatic increases in error for the first 24 hours, but ends up nearly doubled at the end of the forecast horizon of 48 hours.

## 6 Conclusions

In this paper, we have shown that kernel machines provide good mechanisms to create a wind electricity power generation forecasting system. The proposed

kernel machine has been able to adapt to the wind power patterns. A large improvement is obtained by using multiple weather stations over merely one weather station. We have successfully combined the discretised wind speed predictions from several wind stations to form a high resolution accurate power curve. The average wind direction successfully discriminates wind from different directions, as it lowers the error made by the model. Although the model performs good, the quality of the numerical wind forecasts has a large influence on the quality of the wind power predictions.

Future work could include taking into account more weather stations and more variables such as air pressure, humidity, and wind direction at each weather station. Structure in the errors of the numeric weather forecasts could be taken into account. A probabilistic type of kernel machine can give the advantage of estimated confidence intervals of the forecasted wind power production.

## References

- [1] Mark Aizerman, Emmanuil Braverman, and Lev Rozonoër. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, 25:821–837, 1964.
- [2] HM Al-Hamadi and SA Soliman. Short-term electric load forecasting based on kalman filtering algorithm with moving window weather and load model. *Electric Power Systems Research*, 68:47–59, 2004. ISSN 0378-7796.
- [3] Svetlana Borovkova. *Estimation and Prediction for Nonlinear Time Series*. PhD thesis, University of Groningen, 1998. URL <http://dissertations.ub.rug.nl/faculties/science/1998/s.a.borovkova/>.
- [4] Adrian Bors and Ioannis Pitas. Median radial basis function neural network. *IEEE Transactions on Neural Networks*, 7(6):1351–1364, 1996. ISSN 1045-9227. URL <http://www-users.cs.york.ac.uk/~adrian/Papers/Journals/TNN96.pdf>.
- [5] Martin Casdagli. Nonlinear prediction of chaotic time series. *Physica D: Nonlinear Phenomena*, 35(3):335–356, 1989.
- [6] Harris Drucker, Chris Burges, Linda Kaufman, Alex Smola, and Vladimir Vapnik. Support vector regression machines. In Michael Mozer, Michael Jordan, and Thomas Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, pages 155–161, Cambridge, Massachusetts, USA, 1997. The MIT Press. ISBN 0-262-10065-7. URL <http://www.kernel-machines.org/papers/druburkausmovap96.ps.gz>.
- [7] Mário Figueiredo. Adaptive sparseness using Jeffreys prior. In Thomas Dietterich, Suzanna Becker, and Zoubin Ghahramani, editors, *Advances in Neural Information Processing Systems (NIPS'01)*, volume 14, pages 697–704, Cambridge, Massachusetts, USA, 2002. The MIT Press. ISBN 0-262-04208-8. URL <http://books.nips.cc/papers/files/nips14/AA07.pdf>.
- [8] Eithne Fitzgerald. Directive 96/61/EC of the European parliament and of the council of 24 September 1996 concerning integrated pollution prevention and control. *EU Official Journal*, L(61):26–40, 10 1996. ISSN 0378-6978.

- [9] Nicole Fontaine and Charles Picqué. Directive 2001/77/EC of the European parliament and of the council of 27 September 2001 on the promotion of electricity produced from renewable energy sources in the internal electricity market. *EU Official Journal*, L(283):33–40, 10 2001. ISSN 0378-6978. URL <http://tinyurl.com/y7n9ko>.
- [10] Gregor Giebel, Richard Brownsword, and George Kariniotakis. The state-of-the-art in short-term prediction of wind power: A literature overview. Deliverable report D1.1, Project ANEMOS, Roskilde, Denmark, 2003. URL [http://anemos.cma.fr/download/ANEMOS\\_D1.1\\_StateOfTheArt\\_v1.1.pdf](http://anemos.cma.fr/download/ANEMOS_D1.1_StateOfTheArt_v1.1.pdf).
- [11] Boris Igel'nik and Yoh-Han Pao. Stochastic choice of radial basis functions in adaptive function approximation and the functional-link net. *IEEE Transactions on Neural Networks*, 6(6):1320–1329, 1995. ISSN 1045-9227.
- [12] Nicolaos Karayiannis. Reformulated radial basis neural networks trained by gradient descent. *IEEE Transactions on Neural Networks*, 10(3):657–671, 1999. ISSN 1045-9227. URL <http://tinyurl.com/yzpxkk>.
- [13] Michael Kearns and Umesh Vazirani. *An Introduction to Computational Learning Theory*. The MIT Press, Cambridge, Massachusetts, USA, 1994. ISBN 0-262-11193-4. URL <http://mitpress.mit.edu/book-home.tcl?isbn=0262111934>.
- [14] Neil Lawrence, Matthias Seeger, and Ralf Herbrich. Fast sparse gaussian process methods: The informative vector machine. In Suzanna Becker, Sebastian Thrun, and Klaus Obermayer, editors, *Advances in Neural Information Processing Systems (NIPS'02)*, volume 15, pages 625–632, Cambridge, Massachusetts, USA, 2003. The MIT Press. URL <http://books.nips.cc/papers/files/nips15/AA16.pdf>.
- [15] Shuhui Li, Donald Wunsch, Edgar O'Hair, and Michael Giesselmann. Using neural networks to estimate wind turbine power generation. *IEEE Transactions on Energy Conversion*, 16(3):276–282, 9 2001. URL <http://tinyurl.com/yb58sv>.
- [16] Shuhui Li, Donald Wunsch, Egard O'Hair, and Michael Giesselmann. Comparative analysis of regression and artificial neural network models for wind turbine power curve estimation. *Journal of Solar Energy Engineering*, 123:327–332, 11 2001. ISSN 0199-6231. URL <http://www.ece.umn.edu/acil/Publications/JOURNAL/ASMEJSEE01.pdf>.
- [17] David MacKay. Bayesian interpolation. *Neural Computation*, 4(3):415–447, 1992. ISSN 0899-7667. URL <http://www.inference.phy.cam.ac.uk/mackay/inter.nc.ps.gz>.
- [18] Pierre Pinson, Nils Siebert, and George Kariniotakis. Forecasting of regional wind generation by a dynamic fuzzy-neural networks based upscaling approach. In *Proceedings of the European Wind Energy Conference (EWEC 2003), Madrid, Spain*. EWEA, 2003. URL <http://tinyurl.com/yx3ht2>.
- [19] Bernhard Schölkopf and Alexander Smola. *Learning with Kernels*. Adaptive Computation and Machine Learning. The MIT Press, Cambridge, Massachusetts, USA, 2002. ISBN 0-262-19475-9. URL <http://www.learning-with-kernels.org>.

- [20] Colin Cowan Sheng Chen and Peter Grant. Orthogonal least squares learning algorithm for radial basis function networks. *IEEE Transactions on Neural Networks*, 2(2):302–309, 3 1991. ISSN 1045-9227. URL [http://itswww.epfl.ch/~coursnonlin/files/support/ols\\_rbf.pdf](http://itswww.epfl.ch/~coursnonlin/files/support/ols_rbf.pdf).
- [21] Leslie Valiant. A theory of the learnable. *Communications of the ACM*, 27(11): 1134–1142, 1984. ISSN 0001-0782. URL <http://www.cs.toronto.edu/~roweis/csc2515/readings/p1134-valiant.pdf>.
- [22] Vladimir Vapnik. *Estimation of Dependencies Based on Empirical Data*. Springer series in statistics. Springer-Verlag, New York, 1982. ISBN 0-387-90733-5. Translated from Russian.
- [23] Vladimir Vapnik. *The Nature of Statistical Learning Theory*. Statistics for Engineering and Information Science. Springer-Verlag, New York, 1995. ISBN 0-387-98780-0. URL <http://www.springer.com/east/home?SGWID=5-102-22-2017705-0>.
- [24] Vladimir Vapnik. *Statistical Learning Theory*. John Wiley & Sons, New York, 1998. ISBN 0-471-03003-1. URL <http://www.wiley.com/cda/product/0,,0471030031,00.html>.
- [25] Grace Wahba. Spline models for observational data. *Journal of the Royal Statistical Society. Series B*, 59:133–150, 1990. ISSN 0035-9246.