Stochastic assessment of aerodynamics within offshore wind farms based on machine-learning

M Richmond[a], A Sobey[b], R Pandit[a], A Kolios[a]
[a] *Department of Naval Architecture, Ocean & Marine Engineering, University of Strathclyde, Glasgow, G4 0LZ, United Kingdom*
[b] *Fluid Structure Interactions, University of Southampton, University Rd, Southampton SO17 1BJ, England, United Kingdom*

Corresponding author email: mark.richmond@strath.ac.uk

Abstract. Wind turbine flow field prediction is difficult as it requires computationally expensive computational fluid dynamics (CFD) models. The contribution of this paper is to propose and develop a method for stochastic analysis of an offshore wind farm using CFD and a non-intrusive stochastic expansion. The approach is developed through testing a range of machine-learning methods, evaluating dataset requirements and comparing the accuracy against site measurement data. The approach used is detailed and the results are compared with real measurements obtained from the existing wind farm to quantify the accuracy of the predictions. An existing offshore wind farm is modelled using a steady-state CFD solver at several deterministic input ranges and an approximation model is trained on the CFD results. The approximation models compared are Artificial Neural Networks, Gaussian Process, Radial Basis Function, Random Forest and Support Vector Regression. RBF achieves a mean absolute error relative to the CFD model of only 0.54% and the error of the SVR predictions relative to the real data, with scatter, was 12%. A Jensen model is used for comparison and achieves an error of 16%. This approach has the potential to be used in more complex situations where an existing analytical method is either insufficient or unable to make a good prediction.

Keywords: Artificial Intelligence; Computational Fluid Dynamics; Wind Turbines; Power Prediction

1. Requirement for stochastic analysis

Offshore wind power is growing rapidly. To continue its growth and expansion into other markets the cost needs to be reduced, particularly operational expense which can range from 13.1% to 56.5% of the levelized cost of the electricity [1]. Offshore wind turbines have grown significantly in size over the past few decades, while the average offshore wind turbine installed in 2017 was 5.9 MW there are currently concepts from manufacturers that will generate around 12 MW. Additionally, the size of the offshore wind farms are growing, with more turbines on new sites and expansions of existing sites. This brings new challenges, including in assessing the loads on turbines; but it also brings new opportunities to reduce costs through effective strategies. To aid in this area, new methods need to be developed to more effectively predict the loads on turbines not only at discrete points in time, but over the entire 25-30 year life span of a project.

This is a peer-reviewed, accepted author manuscript of the following paper: Richmond, M., Sobey, A., Pandit, R., & Kolios, A. (2020). Stochastic assessment of aerodynamics within offshore wind farms based on machine-learning. Renewable Energy, 161, 650-661. https://doi.org/10.1016/j.renene.2020.07.083

A wide range of methods exist for modelling the wake of turbines within a wind farm, from simple engineering approximations to full computational fluid dynamics (CFD). The Navier-Stokes CFD solvers range from models where the turbine is treated simply as an actuator disk (AD), [2], [3], to more computationally intensive models of fully resolved wind turbines, [3], [4]. The fully resolved methods can provide accurate answers and predict the flow to a relatively high accuracy but are limited to modelling only relatively short time intervals due to their computationally intensive nature. For evaluating wind load variation within a wind farm, the actuator disk approach is a good compromise between engineering models and fully resolved rotors. The method is mature, at least 30 years old, with a variety of implementations. Studies have shown that actuator disk produces good agreement with measurements regarding wake deficits, [5], [6]. However, it is hard to cover the wide range of variables experienced by a wind farm through CFD; the wind speed can range from a turbine's cut-in speed of around 3-4 m/s to its cut-out speed at around 24-25 m/s and can flow from all directions. The pitching strategy employed results in a non-linear response and there are a number of parameters that must be accounted for as they affect turbulence intensity, yaw misalignment, blade rotational speeds, wind shear and temperature gradients, which creates a stochastic environment.

Monte-Carlo simulations can be used to account for this stochastic environment to generate a frequency-based probabilistic view [7]. However, these simulations require rapid modelling techniques such as the engineering wake models [8] but these are limited and cannot be used to evaluate time-variant phenomena and are inaccurate when modelling wake redirecting or meandering. Therefore, a more adaptable stochastic method is needed which can not only predict simple cases but also which can be adapted to transient and less conventional problems.

This paper develops a method which combines CFD model results with approximation models in order to build a computationally efficient model of wind values at turbine locations. The purpose of this is that the model could then be used with stochastic inputs to build a probabilistic model of the wind farm. To accomplish this, several CFD cases are run across the range of variables of interest, information of interest is extracted from these cases and used to train supervised machine learning models. From the resulting machine learning models, any combination of the previously varied input variables can be explored, with accurate replication of the CFD model but with significantly lower computational cost; facilitating probabilistic methods. The challenge in applying the machine learning model is to build a suitable approximation model for interpolating between training data points, which are individual numerical values in this case. There are a wide range of potentially suitable models for this task which can be used in a regression or surrogate approach; five state-of-the-art methods are chosen for this study: Artificial Neural Networks (ANN), Gaussian Process (GP) with a Radial Basis Function kernel, Radial Basis Function (RBF), Random Forest (RF) and Support Vector Regression (SVR). These methods are compared in both their ability to replicate CFD data after training, as well as the overall accuracy in predicting the wind flow values within a reference wind farm. In addition, several different methods of generating the training data from the CFD data are also compared. This results in a novel method for stochastic analysis of wind turbine loading within a wind farm, as it is now possible to produce results which account for three-dimensional effects in a computationally rapid manner. This approach combines the ability to capture the physics of the problem to a suitable level using CFD analysis, combined with the computational efficiency of machine learning methods to be able to conduct a stochastic analysis.

In section 2, the reference offshore wind farm and the reference data available are presented. In section 3 the CFD model, previously verified in [9], is summarised. Section 4 presents the machine learning methods that are benchmarked. Results are presented in sections 5. A discussion of the findings is given in section 6 and finally, conclusions are given in section 7.

2. Description of the reference offshore wind farm

A series of CFD analyses are conducted for a discrete set of variables, in this case different wind directions and freestream velocities. The results from these analyses are then used to train machine learning models, from which predictions for other values can be generated. The trained approximation models can then be used to replicate the CFD, especially useful in computationally expensive simulation methods like Monte Carlo Simulations, Particle Swarm Optimisation and Genetic Algorithms.

A real offshore wind farm is simulated using actuator disks in a CFD domain based on weather data collected from the farm and the SCADA data collected at four turbine locations. The turbines have a hub height of 80 m. A meteorological mast (met mast) is permanently positioned on the edge of the site, shown on the reference map of the wind farm in Figure 1. The met mast has various sensors which can record wind speed and directions at heights of 25, 30, 58, 72 and 82 m. There are also sensors that record barometric pressure and temperature at 20 m as well as relative humidity and temperature at 82 m. These data are recorded as averages over 10-minute intervals, along with a time-stamp for each sample.

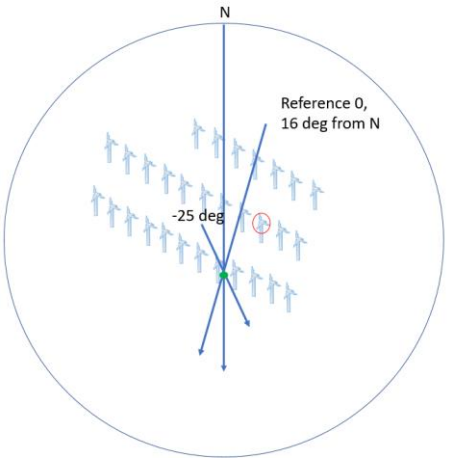

Figure 1. *Map of the modelled wind farm, turbine 16 is circled in red, velocities at Turbine 16 are predicted in later sections. Blue arrow shows wind directions at 16 degrees and -25 degrees for reference.*

3. CFD simulation of the wind farm

The purpose of the CFD model is to generate training data for the machine learning techniques, replicating the real turbines on the farm. The results of the predictions from the CFD-ML approach are compared to real data to evaluate the predictive capabilities.

The CFD model used is a steady-state, Reynolds Averaged Navier-Stokes (RANS) model in the opensource solver OpenFOAM using the SIMPLE solver algorithm. In this approach the three-dimensional wind farm domain is divided into roughly 60 million cells and discretised form of the Navier-Stokes equations are solved iteratively across the domain. The turbines are modelled using actuator-disks which impart a momentum loss, both axially and tangentially, to the fluid flow. The performance of each turbine is controlled individually within the solution process. The CFD results used in this paper were from the k-ε turbulence model. The k-ε model averages the turbulence fluctuations and solves the quantities of k, turbulence kinetic energy, and ε, turbulence dissipation rate. Full details about the CFD model, how it is constructed and an accuracy comparison to real data, can be found at [9].

The wind farm is modelled at a range of incoming wind angles between 35° and -35°, shown in **Table 1,** and the freestream velocity is modelled between 5 m/s and 15 m/s in steps of 2.5 m/s; all combinations of these velocities and angles are modelled with the CFD analysis, resulting in 65 cases. These 65 cases comprise the full dataset. A reduced dataset of only 35 samples was also used to investigate the suitable number of samples. 13 additional points are also included at zero wind speed and power, to reduce the region of extrapolation for the models and ensure a good fit at lower wind speeds. There are 13 points as they are one for each wind direction.

The values for the incoming wind angle are chosen as the accuracy is compared with values from the met mast and this range includes several different wake behaviours: including the met mast being directly behind two turbines and being partially in the wake of close neighbouring turbines. These differences should result in variability in the real values and in the model's ability to predict them.

***Table 1**. Independent variable values from which all CFD cases were based.*

List of Directions (degrees)	-35	-27.5	-20	-15	-10	-5	0	5	10	15	20	27.5	35

The training samples from the CFD model are compared to the wind farm data samples to determine their accuracy by comparing predicted velocity values to those at the met mast, shown in **Figure 1**. The values compared, due to the availability of the data, are the mean velocity deficit value at 10 minute intervals. The mean absolute error is 0.85 m/s, or 8.94% relative to the comparison values. This accuracy should be kept in mind when evaluating the accuracy of the ML predictions. Conducting the CFD analysis is also computationally demanding, even for a steady state analysis. Each of these cases required around 22 hours using 32 CPU cores at a time. On this system, 128 cores can be used simultaneously, so 4 cases can be solved at once. This CFD-ML approach is then only feasibly if there is access to large computational capabilities.

4. Machine learning methods

Several approximation methods are tested to evaluate their applicability as rapid surrogates for the CFD. The effectiveness of these predictions is compared with additional CFD test cases. The models compared are: Artificial Neural Networks (ANN), Gaussian Process (GP), Radial Basis Function (RBF), Random Forest Regression (RF) and Support Vector Regression (SVR). These models were selected because they are in common use within wind energy [10], [11].

Due to the CFD cases requiring so much computational resource, only a few cases can be run and so only a small number of samples can be generated. The model is steady state and the site data available is mean velocity values at 10-minute intervals, and so each case relates to one value. The number of samples used to train the approximation method is an important consideration, it's important to determine how much data is required. While regression methods typically perform better with hundreds or even thousands of data samples, surrogate models can perform well with far fewer, but tend to overfit with too many sample points. For example, Queipo et al. [12] performed a case study using 54 designs to optimise 4 parameters using a surrogate model based on radial basis function (RBF). In another example using RBF [13], 200 samples were used to estimate soil parameters in a 400 km by 700 km area, finding this number of training samples suitable. For the response surface method, for a full factorial design of experiments, it is recommended to have a minimum of either 2^N samples or 3^N samples including center points, where N is the number of variables [14], this is more of an absolute minimum. Realistically, this depends on the situation and often 10^N is more suitable. The following study uses 65 training sample points to model the 2 variables.

4.1. Artificial Neural Networks

Artificial Neural Networks (ANN) were first coded by Rosenblatt as a 'perceptron' in 1957 [15], although the idea had been considered in physical models earlier than this [16]. Neural networks are widely used in wide range of applications including wind energy in both condition monitoring, [17], [18], for damage detection [19]–[21], load forecasting [22], to infer loads from standard signals including yaw misalignment, generator speed and electrical power [23] and in power production and resource estimation [24]–[27]. Due to the power and versatility of ANNs, they have found wide spread use [10].

Multi-Layer Perceptron Neural Networks employ several layers of neurons: an input layer, one or more hidden layers and an output layer, as shown in **Figure 2**. The neurons have activation functions to relate the input they receive to the output they send to the next layer. In addition, there are weights between different neurons and biases which are trained iteratively, traditionally using reinforcement learning through back-propagation where the weights are adjusted between each neuron based on how accurate the prediction is to the required result. The model can be used in an approximation problem by using an appropriate output layer function and minimizing a cost function.

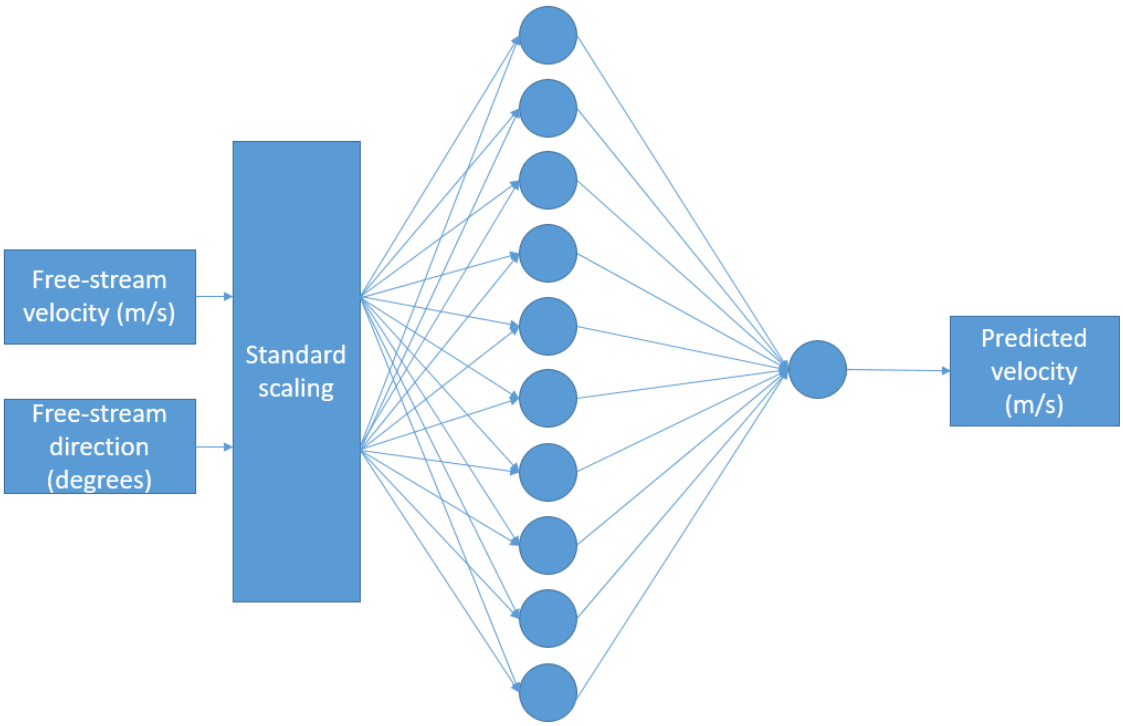

Figure 2. *Architecture of the implemented ANN*

Results were obtained using Keras with a TensorFlow backend. This model requires a large number of epochs to acheice suitable convergence, and hence a long training time. A minimum of 25,000 epochs are required with the best results at 60,000. After hyperparameter tuning the best results were obtained using: 10 nodes for the hidden layers, 'elu' activation for the hidden layer and 'linear' activation for the output node. Only one hidden layer was used as more were not necessary to capture the features. The system was trained using the 'adam' optimizer and the 'mean_square_error' loss parameter.

4.2. Gaussian Process Regression

Gaussian process regression, or kriging, is a stochastic process where a distribution is defined over a basis function which can take any form, in this case a squared exponent function. One of the earliest instances of a gaussian process being used for curve fitting is from O'Hagan [28]. The methodology is as follows: after obtaining a predictive distribution, the regression is then applied over a basis function which projects the input onto the feature space, using a Bayesian approach to take a prior distribution and updating it to form a posterior distribution [17-19]. The output of the Gaussian Process is defined as the mean and covariance matrix, the covariance matrix defining the smoothness and can be represented by the kernel function. Given a set of independent variables, x, Gaussian process can be fully defined with a mean, m(x), and a covariance, k(x, x'), as given in equation 1 [29];

$$\mathcal{F}(x) \sim GP(m(x), k(x, x')). \tag{1}$$

Gaussian Process has been used in wind energy for wind power forecasts [32]–[35] as well as determining wind turbine performance and efficiency [36] and condition monitoring through SCADA signal assessment [11], [37], [38] where the ability to determine confidence bounds in the prediction has been useful.

In the results presented below the algorithm is implemented in python using the Scikit learn library, *sklearn.gaussian_process.GaussianProcessRegression* using a radial basis function kernel [39]. This implements a GP model using algorithm 2.1 by Rasmussen and Williams [29]. A useful feature here is

that it allows a prediction without requiring a fit for the prior. In this study, a squared exponential kernel is used which is parameterised by a length scale, l, which in this case was set at 0.1. This kernel is given in equation 3 [40];

$$k(x_i, x_j) = \exp\left(-\frac{1}{2} d\left(\frac{x_i}{l}, \frac{x_j}{l}\right)^2\right). \quad (2)$$

4.3. Radial Basis Function

A typical Radial Basis Function (RBF) model, originally derived by Broomhead and Lowe [41], is a form of feed-forward neural network composed of single neurons using radial basis function transfer functions [42]–[44]. The result of this approach is that an radial basis function fits a surface through the measured sample points. The values between the sample points are determined from functions based on the radial distance from the original point. The method has been used in soil parameter estimation [22, 23] and recently in wind energy to create a wind turbine controller which can deal with system nonlinearities [47] as well as in a LIDAR based controllers [48].

The equation for a multi-quadratic basis function is given in (3) [49],

$$\phi(r) = \sqrt{1 + (\varepsilon r)^2} \quad (3)$$

where ε is the shape parameter and r is the radius. The output from the entire model can then be given in equation 5, where the basis-function is here shown as a function of the Euclidian norm which is effectively the radius shown previously.

$$s(x) = \sum_{i=1}^{n} \lambda_i \phi(\|x - x_i\|) \quad (4)$$

where s(x) is the output of the model and λ_i is is the weight of the i-th node.

RBF nodes can be applied in a variety of models, however they are traditionally applied in a single-layer model with a single node for each sample point in the training set, the structure of which is shown diagrammatically in Figure 3.

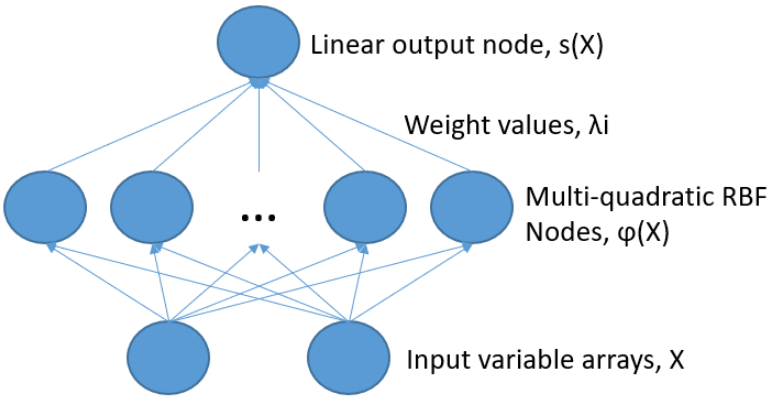

Figure 3. RBF schematic diagram

The code was developed in Matlab to apply a multi-quadratic radial basis function. The code iteratively optimises the shape parameter, ε. As the RBF method forces the interpolation at the sample points, a least-squares method is not applicable and therefore LOO cross-validation is used to assess accuracy.

4.4. Random Forest Regression

This method is more recent and was first proposed by Ho [50] in 2001 and improved by Breiman [51]. Random forest regression is an ensemble method based on Decision Tree Analysis where the results of

multiple Decision Tree models are averaged to produce a prediction. The key idea is that for multiple noisy, but mostly unbiased models, accuracy can be improved by reducing variance. The forest of trees is as accurate as the trees from which it is made, but the combination can reduce variance. To best accomplish this, trees must be generated in such a way as to minimise the correlation between trees while maintaining low variance of individual trees. The variance of the forest of B identically distributed trees can be shown in equation 6 [52],

$$\rho\sigma^2 + \frac{1-\rho}{B}\sigma^2, \tag{5}$$

where σ^2 is the variance of the trees and ρ is the correlation between individual trees. To accomplish this low correlation and low variance of trees, the random forest process randomly selects variables, or sets of variables, to use to form the splits at nodes. The distinction between RF classification and RF regression is simply that for classification the trees cast votes for the prediction while in regression an average is taken.

The method is widely used in some software applications because it is fast and robust, for example it was used by Microsoft for pose recognition in their Xbox Connect [53]. However the method has seen only limited use in wind energy; RF classifiers have been used to predict blade icing [54] and wind turbine stoppages [55] but so far have received limited attention in wind energy applications.

The algorithm is implemented in python using the Scikit learn library, *sklearn.ensemble.RandomForestRegression* [56]. Implementing this algorithm, the parameter changed is the number of trees, which in this case was set at 10,000. The criteria for the quality of split was mean-squared error. There are other parameters which can be set in order to avoid 'fully-grown' and 'unpruned' trees to save on memory, however with the size of the dataset this wasn't an issue and these parameters were set at their default as defined in the documentation.

4.5. Support Vector Regression

Support Vector Regression (SVR) is where support vector machines (SVM) are used for regression. SVMs, first identified by Vapnik [57], work through linear domain division where the division is made to be as large as possible. This can also be extended to higher order domains and be used for regression through the use of kernels [19, 20]. In this method, a support vector is drawn such that the error is minimized by selecting a hyperplane which maximises the margin. For linear SVR, an approximation can be derived from equation 7, [60];

$$y = \sum_{i=1}^{N}(\alpha_i - \alpha_i^*) * \langle x_i, x\rangle + b, \tag{6}$$

where α_i and α_i^* are Lagrange multipliers, there are N training variables and b is a real number constant. For non-linear SVR, a kernel is applied to $\langle x_i, x\rangle$. For a Gaussian radial basis function kernel this becomes equation (7), where σ is a free parameter,

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right). \tag{7}$$

Support vector machines can be applied in either regression or classification form and have been used in power forecasting for wind energy [61]–[63], where it was found that accurate predictions can be made and that hybrid methods can improve predictions. SVM has also been used in fault-detection [64], [65] with good performance, but also relatively quick training times compared to other methods tested due to their simplicity.

The algorithm is implemented in python using the Scikit learn library, *sklearn.svm.SVR* using a radial basis function kernel [66]. An RBF kernel was used in this case. The two tuneable parameters for this SVR implementation with RBF kernel are the penalty parameter, C, and epsilon-tube where no penalty is associated, ε, which were optimized for this data with values of 7 and 0.05 respectively.

5. Results

The predictions from the approximation models, trained to the CFD data, are compared against measured wind speed values from the real wind farm. A total number of 2016 data samples recorded from the met mast recorded data are taken and predicted by the CFD-approximation process. These samples are 10-minute average velocity values. The met mast values are measured at 82 m above sea level and the CFD results for comparison are extracted at this same coordinate in the model.

For each of the 2016 met mast recorded samples, there are corresponding free-stream conditions which were recorded at the same time; these free-stream conditions are used as independent variables in the approximation models to predict the values at the met mast. However, there is a slight mismatch between the free-stream conditions recorded at the time and the values recorded at the met mast as they are from different sources and matched by their timestamps. In addition, large scale turbulence makes the data stochastic, resulting in variations of the flow. Therefore, a second comparison is made between the already trained approximation model and the CFD results at the met mast for seven new free-stream conditions. These new CFD cases are modelled with free-stream conditions shown in **Table 2** and were chosen with a strategy to position them roughly midway between free-stream conditions previously used to generate training data. The 'free-stream' values are measured at 80 m and the met mast values are measured at 82 m, however this difference in height is consistent for both the CFD model as well as the measured data and so should not lead to a difference in this comparison.

Table 2. *Independent variables for the three CFD comparison cases*

	1	2	3	4	5	6	7
Wind Direction (degrees)	0.00	12.50	12.50	7.00	18.75	6.25	18.75
Freestream Velocity (m/s)	8.25	10.00	8.25	6.50	11.75	13.50	6.50

5.1. Artificial Neural Networks

Figure 4 shows a Y-Y plot of recorded met mast velocities against predictions from the combined CFD-ANN approach, for 100 randomly selected cases to predict, with the metrics calculated for 2016 total number of samples. The figure shows points for when both the 35 training samples and 65 training samples are used as well as when a 0 free-stream velocity case is used to reduce extrapolation.

Additionally, the mean absolute errors are shown in the legend. The 45° line represents the points where the x-axis values are equal to the y-axis values, that the error is 0.

Including the zero reduces the deviation from the real values at lower velocities slightly. This is reflected in the mean absolute error (MAE) values which are slightly lower for the 'with 0' cases than without. Although this changes the mean bias error (MBE), it's not consistently one way or the other. Using 65 training samples rather than 35 also improves the MAE, although this can't be seen in the scatter plot because the difference is slight. The CFD predictions are, erroneously, a little lower than the met mast values at 12.5 m/s. These MAE values are consistently the second lowest of the approximation methods, very closely behind SVR.

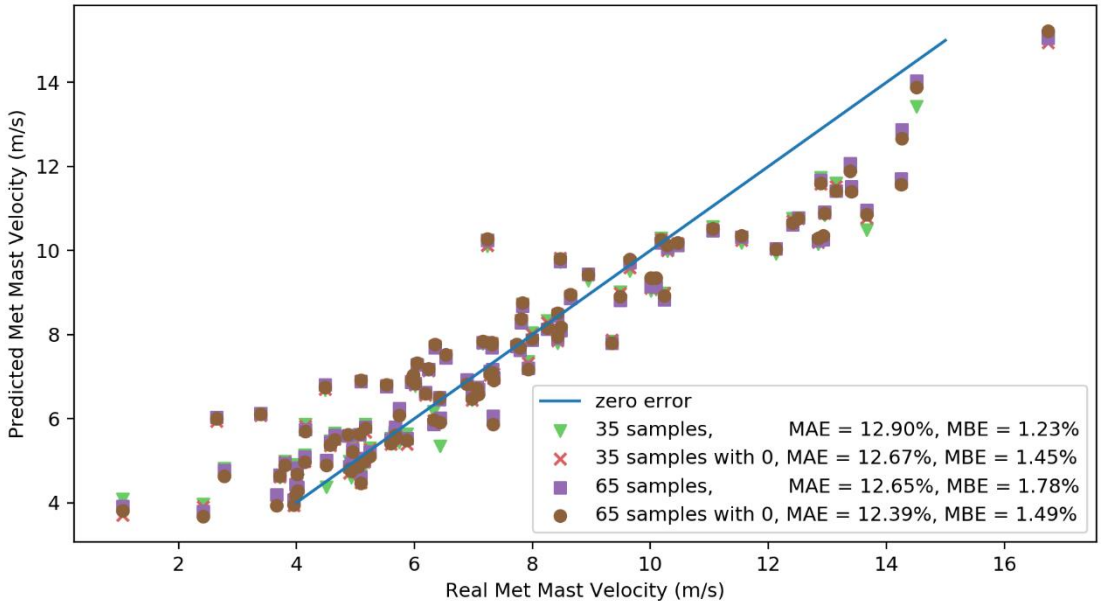

***Figure 4**. ANN, plotted 100 predicted cases, real velocity (m/s) against predicted velocity (m/s). The legend shows the training dataset used for the predictions as well as error metric calculated using 2016 total samples, including the mean absolute error, MAE, and mean bias error, MBE, achieved by that model.*

5.2. Gaussian Process

Figure 5 shows a Y-Y plot of recorded met mast velocities against predictions from the combined CFD-GP approach, for 100 randomly selected cases to predict, with the metrics calculated for 2016 total number of samples. The figure shows points for when both the 35 training samples and 65 training samples are used as well as when a 0 free-stream velocity case is used to reduce extrapolation.

Additionally, the mean absolute errors are shown in the legend. The 45° line represents the points where the x-axis values are equal to the y-axis values, that the error is 0.

At the lower values, where the met mast value is less than 4 m/s, the predicted values are all too high, despite otherwise being close to the 45° line. When training sets which include 0 values are used, then this behaviour is less of an issue, however it is still present. This is reflected in the MBE values which reduce when training sets including the 0 values are used. Surrogate models predict values which pass through the training points and so they are effectively pinned at those points. If values being predicted are outside of the training set, then they are not bound by training values and are free to fluctuate. By adding the 0 values into the training set, this helps pin the surrogate model, extending its applicability.

This model makes predictions which pass through the training set; however it is free to fluctuate away from the training data. Above 5 m/s the training data is spaced by 2.5 m/s intervals, however there is a 5 m/s gap to 0 m/s, this is apparently too large.

The decrease in MAE from the 35 samples training set to the 65 samples training set is the largest of any model. This is hard to tell from the scatter plot, there's nowhere that the larger training set results in significantly better results, so the improvement is throughout a lot of prediction points. The added 0 values make a large improvement in the MAE for reasons already discussed.

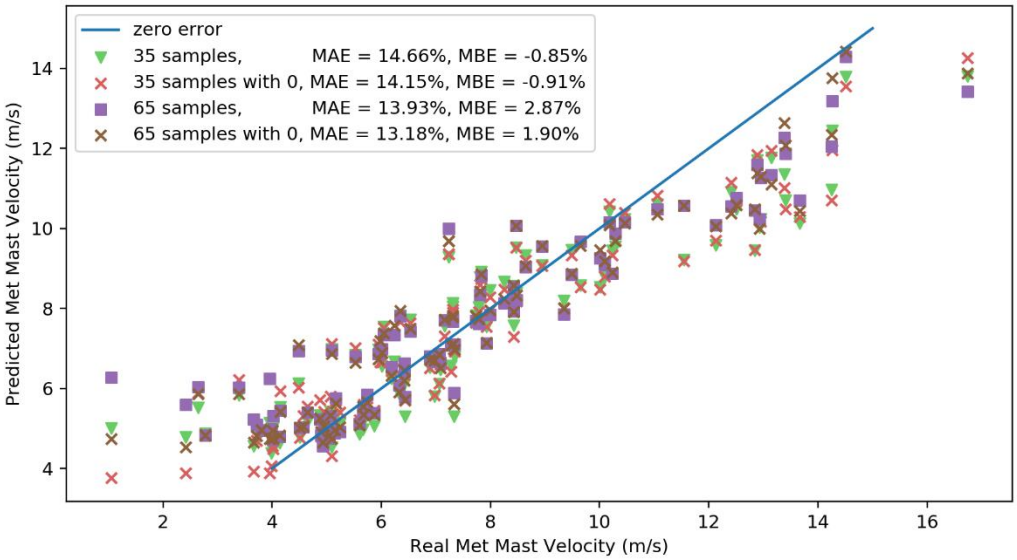

Figure 5. *GP 100 plotted cases, real velocity (m/s) against predicted velocity (m/s). The legend shows the training dataset used for the predictions as well as error metric calculated using 2016 total samples, including the mean absolute error, MAE, and mean bias error, MBE, achieved by that model.*

5.3. Radial Basis Function

Figure 6 Shows a Y-Y plot of recorded met mast velocities against predictions from the combined CFD-GP approach, for 100 randomly selected cases to predict, with the metrics calculated for 2016 total number of samples. The figure shows points for when both the 35 training samples and 65 training samples are used as well as when a 0 free-stream velocity case is used to reduce extrapolation.

Additionally, the mean absolute errors are shown in the legend. The 45° line represents the points where the x-axis values are equal to the y-axis values, that the error is 0.

Although there are a few more points for the 35 training samples model which appear far away from the line, there isn't any significant difference in model accuracy with different training sets, the MAE is consistently close to 13%. As will be shown in section 5.7, there is a large difference in accuracy compared to further CFD cases when the RBF model is trained with more training data, which is not observed in comparison to the met mast. There appears to be a lower limit of MAE which is limited by the accuracy of the CFD to the met mast, rather than by the approximation method to the CFD, and that is just under 12%.

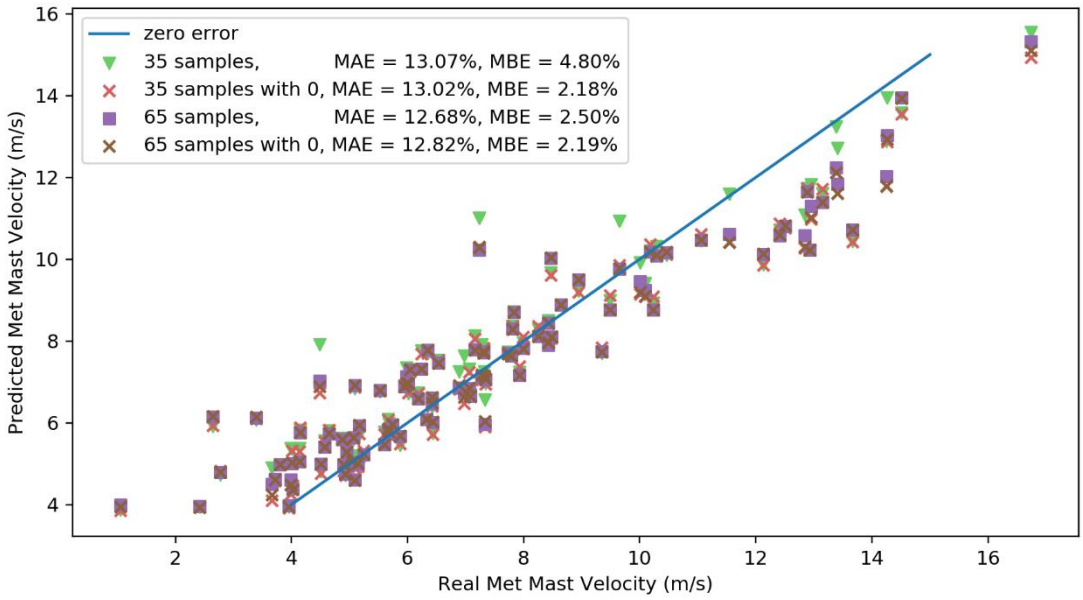

Figure 6. *RBF 100 plotted cases, real velocity (m/s) against predicted velocity (m/s). The legend shows the training dataset used for the predictions as well as error metric calculated using 2016 total samples, including the mean absolute error, MAE, and mean bias error, MBE, achieved by that model.*

5.4. Random Forest

Figure 7 shows a Y-Y plot of recorded met mast velocities against predictions from the combined CFD-RF approach, for 100 randomly selected cases to predict, with the metrics calculated for 2016 total number of samples. The figure shows points for when both the 35 training samples and 65 training samples are used as well as when a 0 free-stream velocity case is used to reduce extrapolation.

Additionally, the mean absolute errors are shown in the legend. The 45° line represents the points where the x-axis values are equal to the y-axis values, that the error is 0.

The MAE values for the RF model is high compared to other models, at around 14.7%, and this is due to the overfitting displayed as clusters are formed around the training data. Because of this overfitting, there is almost no difference between the main four training sets. By using the 'artificial samples' training set the MAE is reduced by nearly two percentage points and becomes in line with the other models. From the scatter plot the overfitting is resolved with the 'artificial samples' training set as the points now no longer form discrete clusters. The mean bias error of RF is the lowest of any of the models.

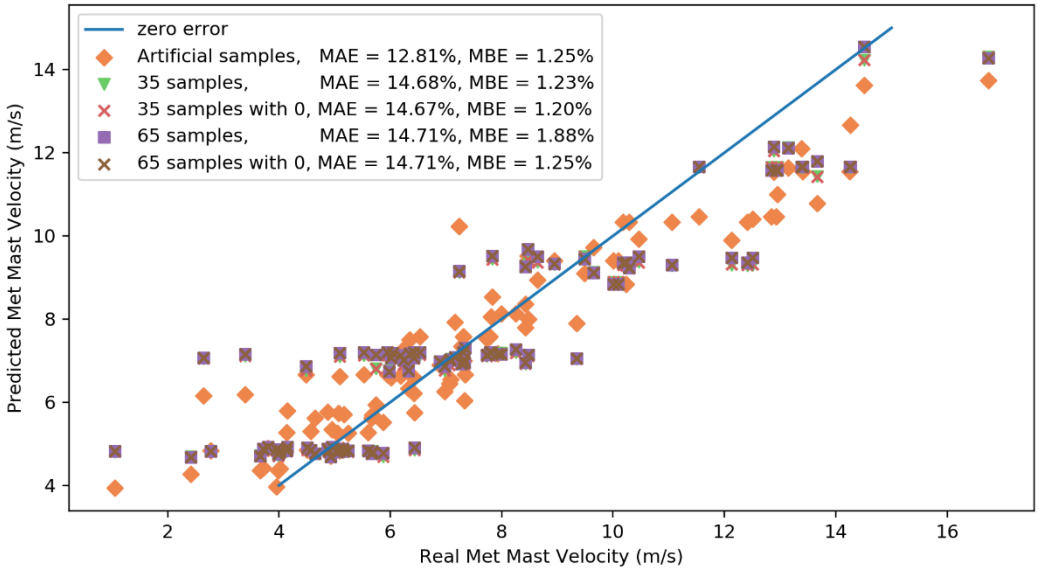

Figure 7. *RF 100 plotted cases, real velocity (m/s) against predicted velocity (m/s). The legend shows the training dataset used for the predictions as well as error metric calculated using 2016 total samples, including the mean absolute error, MAE, and mean bias error, MBE, achieved by that model.*

5.5. Support Vector Regression

Figure 8 shows a Y-Y plot of recorded met mast velocities against predictions from the combined CFD-SVR approach, for 100 randomly selected cases to predict, with the metrics calculated for 2016 total number of samples. The figure shows points for when both the 35 training samples and 65 training samples are used as well as when a 0 free-stream velocity case is used to reduce extrapolation.

Additionally, the mean absolute errors are shown in the legend. The 45° line represents the points where the x-axis values are equal to the y-axis values, that the error is 0.

As can be seen from the MAE values, there's not much difference between the training data sets, except that the training data which included the 0 values are 0.5% points lower than without. The reason can be seen at the lower end of the plot where the points marked with an 'x' are closer to the 45° line than the other points. Without the 0 points the model is extrapolating. Between the 35 training data points to 65 training points, there is only a very minor improvement in MAE. There's so little difference in the 35 training and 65 training models that the predictions are on top of each other and the green triangles are not visible.

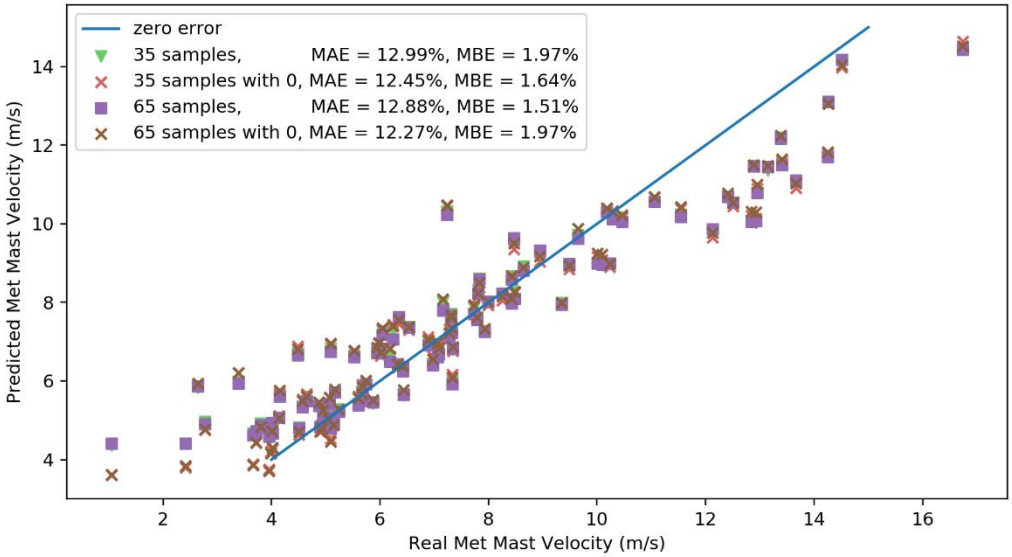

Figure 8. *SVR 100 plotted cases, real velocity (m/s) against predicted velocity (m/s). The legend shows the training dataset used for the predictions as well as error metric calculated using 2016 total samples, including the mean absolute error, MAE, and mean bias error, MBE, achieved by that model.*

5.6. Model results with direction

Figure 9 and ***Figure 10*** show the model behaviour along the range of directions for a given free-stream velocity for the 35 training sample dataset and the 65 training sample dataset respectively. In the physical system being modelled, there is a relation between wake deficit at the met mast and the inflow direction because the wakes of the turbine follow wind direction and so the MM will move in and out of turbine wakes. This physical behaviour is captured in the CFD and hence the blue dots in the following figures, indicating CFD result values, are not all at the same value. The purpose of the machine learning approaches trained on the CFD results is to approximate this behaviour in values between the CFD values.

The predictions for Radial Basis Function and Gaussian Process, which uses a Radial Basis Function kernel, both pass through all the CFD sample points as they utilise these points within their prediction. Support vector regression complies loosely with the training sample points, although the prediction dips slightly around 0 degrees, it doesn't dip as much as low as the CFD prediction. The Artificial Neural Network model performs very well based on these figures, the predictions pass closely to the CFD training points and does not show any strange or excessive behaviour. Of the regression models, RF appears to perform the best; RF captures the general behaviour of the direction change relatively well however it does not reach the lowest point at the 0-degree direction. An advantage of the RF method is that it is unlikely to produce extreme, unrealistic results as it takes the average of decision tree predictions, while some of the other methods can produce extreme, unrealistic results. In that regard, RF is the most robust of the methods tested, but not necessarily the most accurate.

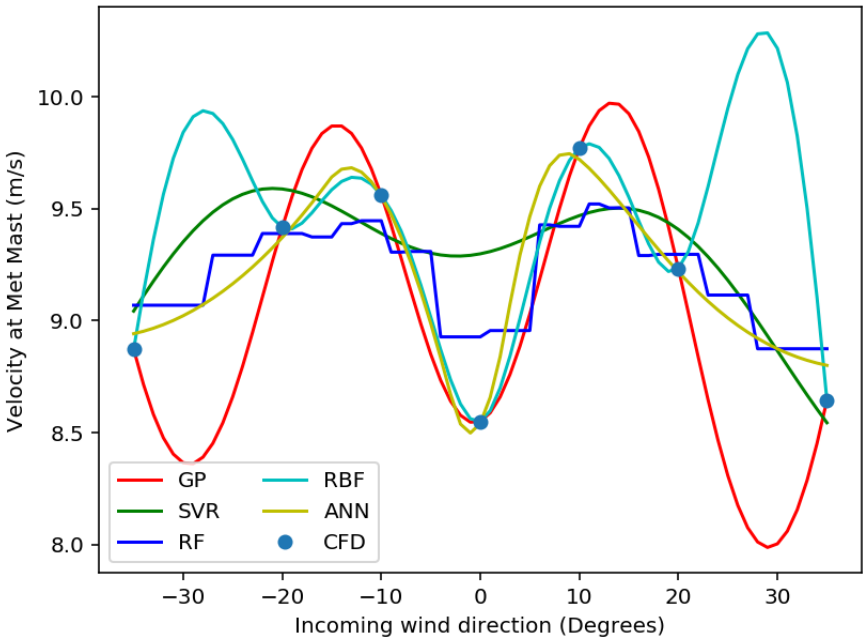

Figure 9. Reduced CFD dataset of 35 cases, *Model prediction for the met mast velocity at different free-stream velocities, k-ε, 15 m/s,*

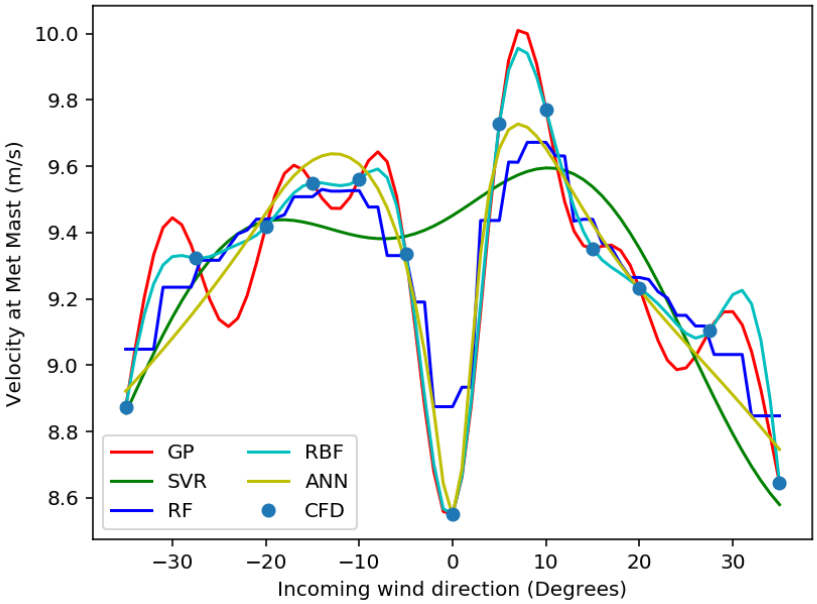

Figure 10. *Full CFD dataset of 65 cases, plots of model prediction for met mast velocity against direction for a given free-stream velocity,* k-ε, 15 m/s,

5.7. Comparison to CFD model

The following charts show a comparison of the predictions from the approximation models with the CFD model results. In this case, seven further CFD cases were conducted and the approximation models were used to predict the results in order to get an estimate for what level of error comes from the ML approximation of CFD. The MAE as a percentage of the mean CFD value are shown in Figure 11 and the maximum errors are shown in Figure 12.

Additionally, a sensitivity study was conducted where the number of training samples was increased from the minimum of 35 cases to the maximum of 65. These two additional steps were using 55 and 45 training samples. To arrive at 55 samples, the results at the direction of +/- 27.5º were removed and for 45 samples, results with direction +/- 15º were removed from that.

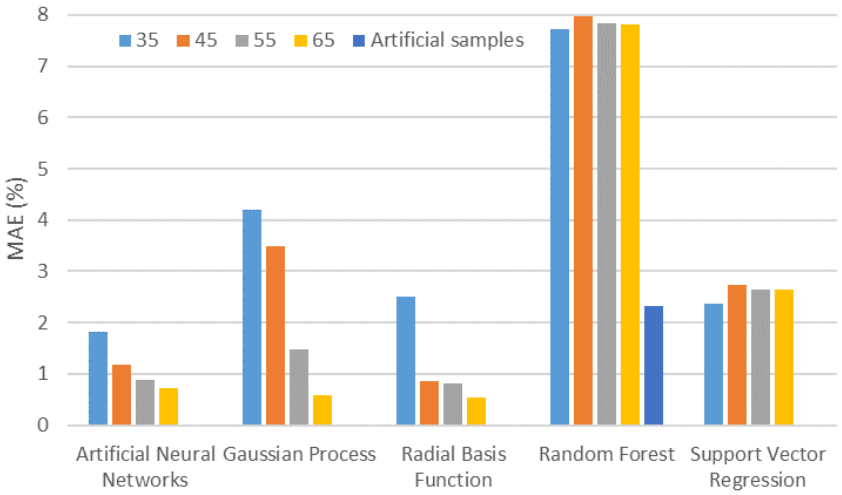

Figure 11. MAE as a percentage of the mean, for each approximation method compared to CFD model

The MAE values range from 0.54% for RBF with the 65-training point data set to 7.72% for RF with the 35-training points data set.

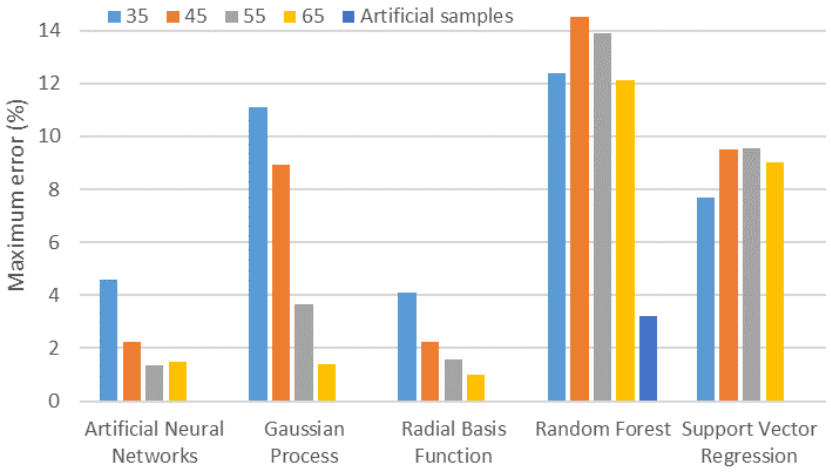

Figure 12. Maximum error percentage, for each approximation method compared to CFD model

The maximum error values range from 1.0% for RBF with the 65-training point data set to 12.4% for RF with the 35-training points data set.

In these results the maximum error is typically twice the MAE with the exceptions of the 'artificial samples' training set for RF where the maximum is slightly less, and for SVR where the maximum is roughly three times the MAE.

5.8 Comparison to established method Jensen

It can be useful to compare these results to other methods used in literature; however, this comparison can be challenging. In a wind tunnel experiment, analytical models can give very close results to reality, for example Garcia et. al. achieved a 2% error relative to a single turbine in wind tunnel tests [67]. For wind farm measurements, where parameters are far less controlled and there are more complications, the error can be much larger. For example, comparing wind farm efficiency using 4 analytical park models, Shao et. al found a MAE at Lillgrund of between 6.48% and 17.06% and at Horns Rev an error of between 5.81% and 10.73% [68]. Gaumond et. al. found that error level was dependant on an accurate knowledge of wind direction and found errors of 20.9%, 20.9%, 21.7% for Jensen, Larsen and Fuga respectively [69]. It can be difficult to compare these numbers directly, not only because they often are errors for different things but also because the error can be dependent on the wind farm itself as well as factors specific to the data recording and collection. Conducting the CFD for this study it was found that a large contribution of error was from the level of scatter in the data and the CFD model achieved an MAE of around 11% relative to measurement data [9].

For a direct comparison, the Jensen wake model was used to evaluate the same wind farm. This is an analytical model which evaluates the wake deficit and propagation as a function of the distance from the turbine. Wake interaction is accounted for by superposition of the wakes. [70], [71] The same 2016 data samples from the met mast are evaluated and the error metrics achieved in this are given in Figure 13. The bias from Jensen is slightly lower than the ML-CFD approach using ANN, around 0.7%, however the MAE is 4% higher at around 16%. It should be noted that the Jensen model predicts velocity at hub height, which is 80 m, and the met mast records values at 82 m.

The effectiveness of this approach can be visualised by plotting the 10-minute averaged met mast values sequentially as they were observed along with model predictions, forming a time-series plot, as in Figure 13. The model used in this case is ANN with 65 training samples and the added 0 points. The behaviours discussed earlier in the paper can be observed in this plot. Typically, the predictions track closely with the real values. There is deviation between 10 and 13 m/s met mast velocities, as mentioned before this is from the CFD model under predicting in this area. There is also some over-prediction at very low values.

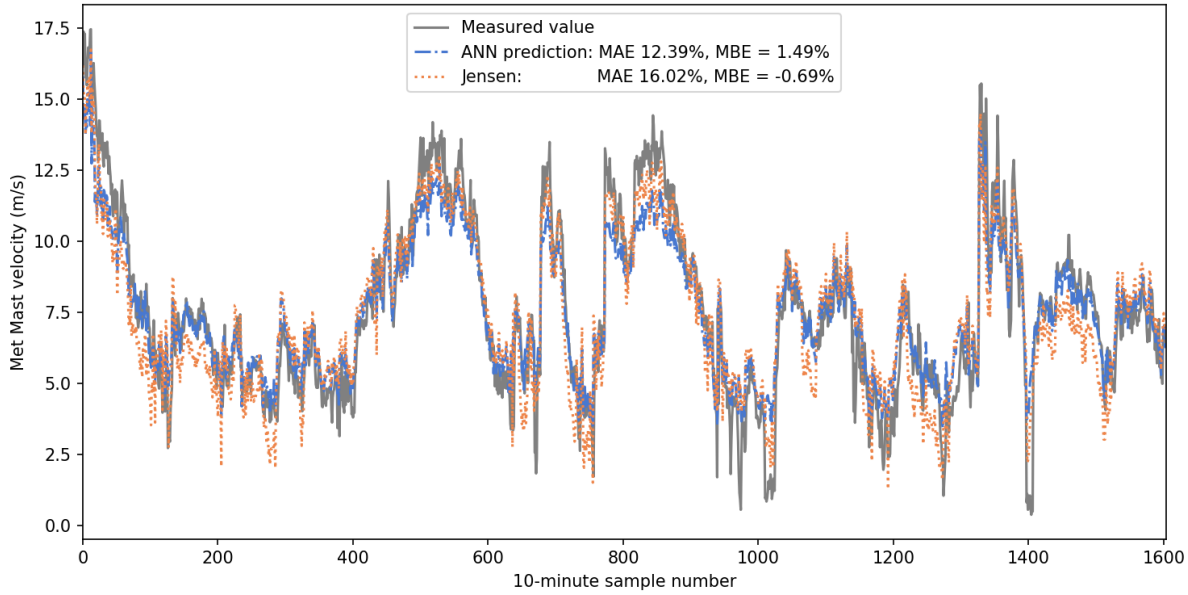

Figure 13. Observations plotted sequentially comparing the met mast values to the predictions from the best method, ANN with 65 training samples and the added 0 points, compared to Jensen wake model.

6. Discussion

The approximation models were trained on CFD generated training data and then used to make predictions which were compared to both further CFD cases as well as against met mast recordings. Against further CFD cases, all methods were able to perform well, with a MAE under 3% for the best training data. However, the methods which clearly performed best against further CFD cases were ANN, RBF and GP as their MAE were all below 1% with the largest training set. In terms of approximating velocity deficit at a given point, these approximation methods can successfully provide similar results as conducting more CFD cases. It should be noted that this approach models the input/output relationship between free-stream values and measured values within the windfarm wake. The approach predicts are individual 10-minute values. If forecasts were to be made, then an appropriate method for statistically sampling free-stream values would need to be used which includes seasonality and cycles.

The goal of the combined CFD-ML approach is to predict the physical system and the accuracy to which this is accomplished is measured by the comparison to met mast results. The CFD model predicts a range of parameters including velocity deficit at various locations as well as turbulence. The ML portion of the approach can then approximate these results to sets of independent variables not used in the CFD models, in this case velocity deficit. The accuracy is compared to measured values recorded on the real wind farm as well as accuracy of the ML models relative to the CFD model. In this study, the lowest MAE was achieved with SVR using the 65 CFD training sample data set with the additional 0 values, with a MAE of only 11.8%. This is despite still having a relatively large error compared to further CFD cases of 2.65% for the 65-point training set. This is because SVR case provides a response which doesn't change as much with direction, while the CFD, based on this, changes too much with direction. It appears from this that the stochastic nature of the real measurement data smears out sharp changes in the result, and so SVR, which approximates without the sudden dip at 0 degrees, achieves a lower error than the other methods against measurement data. Applying this in the future however, it would be good practice to either reduce the error from the CFD or to deliberately account for the stochastic nature of the real data. Otherwise, the best method after SVR is ANN with a lowest MAE of 12.39%. This agrees with the other results, that it is one of the best approximations of the CFD predictions.

Using the larger training set improves the results for the comparison to further CFD cases with all models except for SVR which becomes slightly worse. The improvement from training on the larger data is substantial for comparison to further CFD, reducing MAE to one seventh compared to the smaller training set in one case. The improvement compared to the met mast measurements is not so significant and doesn't even reduce for some models. This is because the error of the approximation models to the CFD is much less than the error of the CFD to the met mast.

Adding the zero values improved the accuracy of almost all cases and can therefore be recommended to be included in future such studies. This provides a noticeable improvement at no computational cost.

The use of artificial samples, effectively using a polynomial regression for one variable before then using the RF, had a big impact in all error metrics. With this training set, the RF model performed as well as the other models. The reason for this is clear in the plot of the randomly chosen cases as the points no longer form discrete clusters due to overfitting.

From a practical standpoint, some of the methods approximate the CFD results to a small error of only around 1% maximum error, but the error to the met mast is much larger. With the use of these approximation methods, similar results to conducting more CFD cases can be obtained at vastly lower computational cost. The overall computational cost of this approach is still high due to requiring so many CFD cases as training data, which in this case required roughly a month on a high-performance computer cluster.

This ML-CFD approach can be useful in cases when measured data isn't available but when a layout is already established. It is more computationally intensive than the Jensen wake model but much more accurate and inherently accounts for physical behaviour of the wake. Therefore, it is useful in cases such as: determining values at points which aren't measured, evaluating different wind farm control strategies or predicting future wind farm performance.

7. Conclusions

In this study, a method is developed for conducting stochastic analysis of an offshore wind farm. This method takes results from CFD analyses at a set of input values, builds an approximation model of the system from the CFD data and produces predictions between the modelled inputs. CFD cases were conducted at a range of free-stream speeds and directions in order to build a suitable training set based on the 65 CFD cases. Velocity deficit at the met mast was predicted by the trained models and was compared to measurement data from the real met mast.

Five Approximation methods were compared: Artificial Neural Networks, Gaussian Process with a Radial Basis Function kernel, Radial Basis Function and Support Vector Regression. These are state-of-the-art methods which all work in very different ways. The suitability and accuracy of each for this task is assessed.

On balance, both Artificial Neural Network and Radial Basis Function were the best method with the most accurate prediction of CFD results, achieving a mean absolute error of 0.73% and 0.54% respectively. Increasing the number of data samples for training had a clear improvement on the replication of CFD results by the approximation methods with the surrogate models, Gaussian Process and Radial Basis Function, showing the largest improvements in replicating the CFD results with an increase in the training data set size. For support vector regression however, there was no improvement with the larger training set.

In the case of comparing against the measured site data, the best method was SVR with a MAE of 12.27%, followed by ANN and RBF which had MAEs of 12.39% and 12.68% respectively. Good practice in future work would be to use either ANN or RBF, which predict CFD results well, and then account for the stochastic nature of the real measurements.

Future work could build on this by incorporating the stochastic capabilities of some approximation models in order to replicate the stochastic nature of the wakes. Additionally, other variables which can be calculated could be approximated such as turbulence values, or a time-series prediction can be generated, which can be useful for estimating fatigue.

Acknowledgements

This work was supported by Grant EP/L016303/1 for Cranfield University, Centre for Doctoral Training in Renewable Energy Marine Structures (REMS) (http://www.rems-cdt.ac.uk/) and by Grant EP/L014106/1, Supergen Wind Hub, both from the UK Engineering and Physical Sciences Research Council (EPSRC). There are no conflicts of interest.

References

[1] T. Poulsen, C. Hasager, and C. Jensen, “The Role of Logistics in Practical Levelized Cost of Energy Reduction Implementation and Government Sponsored Cost Reduction Studies: Day and Night in Offshore Wind Operations and Maintenance Logistics,” *Energies*, vol. 10, no. 4, p. 464, 2017.

[2] F. Porté-Agel, H. Lu, and Y.-T. Wu, “Interaction between Large Wind Farms and the Atmospheric Boundary Layer,” *Procedia IUTAM*, vol. 10, no. 0, pp. 307–318, 2014.

[3] N. Stergiannis, C. Lacor, J. V Beeck, and R. Donnelly, “CFD modelling approaches against single wind turbine wake measurements using RANS,” *J. Phys. Conf. Ser.*, vol. 753, p. 032062, 2016.

[4] L. Wang, R. Quant, and A. Kolios, “Fluid structure interaction modelling of horizontal-axis wind turbine blades based on CFD and FEA,” *J. Wind Eng. Ind. Aerodyn.*, vol. 158, pp. 11–25, 2016.

[5] S. Kalvig, E. Manger, and B. Hjertager, “Comparing different CFD wind turbine modelling approaches with wind tunnel measurements,” *J. Phys. Conf. Ser.*, vol. 555, no. 1, p. 012056, 2014.

[6] B. Johnson, “Computational Fluid Dynamics (CFD) modelling of renewable energy turbine wake interactions,” University of Central Lancashire, 2015.

[7] M. Richmond *et al.*, “Multi-Criteria Decision Analysis for Benchmarking Human-Free Lifting Solutions in the Offshore Wind Energy Environment,” *Energies*, vol. 11, no. 5, p. 1175, May 2018.

[8] M. Gaumond *et al.*, “Benchmarking of wind turbine wake models in large offshore wind farms,” *DTU Wind Energy*, 2012.

[9] M. Richmond, A. Antoniadis, L. Wang, A. Kolios, S. Al-Sanad, and J. Parol, “Evaluation of an offshore wind farm computational fluid dynamics model against operational site data,” *Ocean Eng.*, vol. 193, 2019.

[10] M. Martinez Luengo, A. Kolios, and L. Wang, “Structural Health Monitoring of Offshore Wind Turbines: A review through the Statistical Pattern Recognition Paradigm,” *Renew. Sustain. Energy Rev.*, vol. 64, no. August, pp. 91–105, 2016.
[11] R. K. Pandit and D. Infield, “Comparative analysis of Gaussian Process (GP) power curve models based on different stationary covariance functions for the purpose of improving model accuracy,” *Renew. Energy*, vol. 140, pp. 190–202, 2019.
[12] N. V. Queipo, R. T. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, and P. Kevin Tucker, “Surrogate-based analysis and optimization,” *Prog. Aerosp. Sci.*, vol. 41, no. 1, pp. 1–28, 2005.
[13] C. Rusu and V. Rusu, “Radial basis functions versus geostatistics in spatial interpolations,” *IFIP Int. Fed. Inf. Process.*, vol. 217, no. 1, pp. 119–128, 2006.
[14] R. F. Gunst, “Response Surface Methodology: Process and Product Optimization Using Designed Experiments,” *Technometrics*, vol. 38, no. 3, pp. 284–286, 1996.
[15] F. Rosenblatt, “The Perceptron a Perceiving and Recognizing Automation,” Buffalo, NY, 1957.
[16] W. S. McCulloch and W. H. Pitts, “A logical calculus of the ideas immanent in nervous activity,” *Bull. Math. Biophys.*, vol. 5, pp. 115–133, 1943.
[17] M. Schlechtingen and I. Ferreira Santos, “Comparative analysis of neural network and regression based condition monitoring approaches for wind turbine fault detection,” *Mechanical Systems and Signal Processing*, vol. 25, no. 5. pp. 1849–1875, 2011.
[18] B. Lu, Y. Li, X. Wu, and Z. Yang, “A review of recent advances in wind turbine condition monitoring and fault diagnosis,” *Electron. Mach. Wind*, pp. 1–7, 2009.
[19] N. Dervilis *et al.*, “On damage diagnosis for a wind turbine blade using pattern recognition,” *J. Sound Vib.*, vol. 333, no. 6, pp. 1833–1850, 2014.
[20] E. Gonzalez, M. Reder, and J. J. Melero, “SCADA alarms processing for wind turbine component failure detection,” *J. Phys. Conf. Ser.*, vol. 753, no. 7, 2016.
[21] G. Helbing and M. Ritter, “Deep Learning for fault detection in wind turbines,” *Renew. Sustain. Energy Rev.*, vol. 98, no. September, pp. 189–198, 2018.
[22] N. Dervilis, K. Worden, T. Rogers, D. Scott, and T. Simpson, “Machine Learning for Energy Load Forecasting,” *J. Phys. Conf. Ser.*, vol. 1106, p. 012005, 2018.
[23] U. Smolka, D. Kaufer, and P. W. Cheng, “Are sea state measurements required for fatigue load monitoring of offshore wind turbines?,” *J. Phys. Conf. Ser.*, vol. 555, no. 1, pp. 0–10, 2014.
[24] S. Li, D. C. Wunsch, E. O’Hair, and M. G. Giesselmann, “Comparative Analysis of Regression and Artificial Neural Network Models for Wind Turbine Power Curve Estimation,” *J. Sol. Energy Eng.*, vol. 123, no. 4, p. 327, 2001.
[25] P. Ramasamy, S. S. Chandel, and A. K. Yadav, “Wind speed prediction in the mountainous region of India using an artificial neural network model,” *Renew. Energy*, vol. 80, pp. 338–347, 2015.
[26] D. A. Bechrakis and P. D. Sparis, “Wind speed prediction using Artificial Neural Networks,” *Wind Eng.*, vol. 22, no. 6, pp. 287–295, 1998.
[27] A. M. Foley, P. G. Leahy, A. Marvuglia, and E. J. McKeogh, “Current methods and advances in forecasting of wind power generation,” *Renewable Energy*, vol. 37, no. 1. pp. 1–8, 2012.
[28] A. O’Hagan, “Curve Fitting and Optimal Design for Prediction,” *J. R. Stat. Soc. Ser. B*, vol. 40, no. 1, pp. 1–24, Sep. 1978.
[29] C. E. Rasmussen and C. K. I. Williams, *Gaussian Process For Machine Learning*. 2006.
[30] C. B. Do and H. Lee, “Section notes 9 - Gaussian Processes,” *Lect. notes*, pp. 1–14, 2008.
[31] R. A. Davis, “Gaussian Processes,” *Encycl. Environmetrics Stoch. Model. Environ. Chang.*, p. 6, 2006.
[32] D. Lee and R. Baldick, “Short-term wind power ensemble prediction based on gaussian processes and Neural networks,” *IEEE Trans. Smart Grid*, vol. 5, no. 1, pp. 501–510, 2014.
[33] J. Hu and J. Wang, “Short-term wind speed prediction using empirical wavelet transform and Gaussian process regression,” *Energy*, vol. 93, pp. 1456–1466, 2015.
[34] J. Wang and J. Hu, “A robust combination approach for short-term wind speed forecasting and analysis - Combination of the ARIMA (Autoregressive Integrated Moving Average), ELM (Extreme Learning Machine), SVM (Support Vector Machine) and LSSVM (Least Square SVM) forecasts usi,” *Energy*, vol. 93, pp. 41–56, 2015.
[35] N. Chen, Z. Qian, I. T. Nabney, and X. Meng, “Wind power forecasts using gaussian processes and numerical weather prediction,” *IEEE Trans. Power Syst.*, vol. 29, no. 2, pp. 656–665, 2014.
[36] E. Hart, W. E. Leithead, and J. Feuchtwang, “Wind turbine C p , max and drivetrain-losses estimation using Gaussian process machine learning,” vol. 1037, no. Jounral of Physics: Conference Series, pp. 1–8, 2018.
[37] Y. Li, S. Liu, and L. Shu, “Wind turbine fault diagnosis based on Gaussian process classifiers applied to

operational data," *Renew. Energy*, vol. 134, pp. 357–366, 2019.
[38] R. K. Pandit and D. Infield, "SCADA-based wind turbine anomaly detection using Gaussian process models for wind turbine condition monitoring purposes," *IET Renew. Power Gener.*, vol. 12, no. 11, pp. 1249–1255, 2018.
[39] scikit learn, "sklearn.gaussian_process.GaussianProcessRegression." [Online]. Available: http://scikit-learn.org/stable/modules/generated/sklearn.gaussian_process.GaussianProcessRegressor.html#sklearn.gaussian_process.GaussianProcessRegressor. [Accessed: 22-Feb-2018].
[40] scikit learn, "Radial-basis function (RBF) kernel." [Online]. Available: https://scikit-learn.org/stable/modules/gaussian_process.html#radial-basis-function-rbf-kernel.
[41] D. S. Broomhead and D. Lowe, "Multivariable Functional Interpolation and Adaptive Networks," *Complex Syst.*, vol. 2, pp. 321–355, 1988.
[42] P. Tino, L. Benuskova, and A. Sperduti, *Springer Handbook of Computational Intelligence*, vol. 8, no. 3. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015.
[43] M. J. L. M. Orr, "Introduction to radial basis function networks," 1996.
[44] R. Schaback, "A Practical Guide to Radial Basis Functions," *Sci. Comput. with Radial Basis Funct.*, pp. 1–58, 2007.
[45] Q. Zhu and H. S. Lin, "Comparing ordinary kriging and regression kriging for soil properties in contrasting landscapes," *Pedosphere*, vol. 20, no. 5, pp. 594–606, 2010.
[46] M. Mondrago, "Probabilistic Modelling of Geotechnical Conditions for Offshore Wind Turbine Support Structures," Cranfield University, 2014.
[47] B. N. Roodsari, C. J. B. Macnab, and E. P. Nowicki, "A novel adaptive controller using radial basis function neural network for the wind energy conversion system," *Proc. IEEE Int. Conf. Ind. Technol.*, pp. 715–720, 2017.
[48] B. Han, L. Zhou, and Z. Zhang, "LIDAR-assisted radial basis function neural network optimization for wind turbines," *IEEJ Trans. Electr. Electron. Eng.*, vol. 13, no. 2, pp. 195–200, 2017.
[49] B. Fornberg and C. Piret, "On choosing a radial basis function and a shape parameter when solving a convective PDE on a sphere," *J. Comput. Phys.*, vol. 227, no. 5, pp. 2758–2780, 2008.
[50] Tin Kam Ho, "Random decision forests," in *Proceedings of 3rd International Conference on Document Analysis and Recognition*, 1995, vol. 1, pp. 278–282.
[51] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[52] T. Hastie, R. Tibshirani, and J. Friedman, "Random Forests," in *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2009, pp. 587–605.
[53] J. Shotton *et al.*, "Real-time human pose recognition in parts from single depth images," *Cvpr 2011*, pp. 1297–1304, 2011.
[54] L. Zhang, K. Liu, Y. Wang, and Z. B. Omariba, "Ice detection model of wind turbine blades based on random forest classifier," *Energies*, vol. 11, no. 10, 2018.
[55] K. Leahy, C. Gallagher, K. Bruton, P. O'Donovan, and D. T. J. O'Dullivan, "Automatically Identifying and Predicting Unplanned Wind Turbine Stoppages Using SCADA and Alarms System Data: Case Study and Results," *J. Phys. Conf. Ser.*, vol. 926, no. 1, 2017.
[56] Scikit learn, "3.2.4.3.2. sklearn.ensemble.RandomForestRegression." [Online]. Available: http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html. [Accessed: 22-Feb-2018].
[57] V. Vapnik, *The Nature of Statistical Learning Theory*. Springer, 1995.
[58] P.-H. Chen, R.-E. Fan, and C.-J. Lin, "A study on SMO-type decomposition methods for support vector machines.," *IEEE Trans. Neural Netw.*, vol. 17, no. 4, pp. 893–908, 2006.
[59] S. Suthaharan, *Machine Learning Models and Algorithms for Big Data Classification*, vol. 36. 2016.
[60] A. J. Smola and B. Sch, "A tutorial on support vector regression," Canberra, 2003.
[61] O. Kramer and F. Gieseke, "Short-term wind energy forecasting using support vector regression," *Adv. Intell. Soft Comput.*, vol. 87, pp. 271–280, 2011.
[62] A. Zendehboudi, M. A. Baseer, and R. Saidur, "Application of support vector machine models for forecasting solar and wind energy resources: A review," *J. Clean. Prod.*, vol. 199, pp. 272–285, 2018.
[63] J. Zeng and W. Qiao, "Support vector machine-based short-term wind power forecasting," *2011 IEEE/PES Power Syst. Conf. Expo. PSCE 2011*, vol. 0511, pp. 1–8, 2011.
[64] P. Santos, L. F. Villa, A. Reñones, A. Bustillo, and J. Maudes, "An SVM-based solution for fault detection in wind turbines," *Sensors (Switzerland)*, vol. 15, no. 3, pp. 5627–5648, 2015.
[65] N. Laouti, N. Sheibat-Othman, and S. Othman, "Support vector machines for fault detection in wind turbines," *IFAC Proc. Vol.*, vol. 44, no. 1 PART 1, pp. 7067–7072, 2011.
[66] scikit learn, "sklearn.svm.SVR." [Online]. Available: http://scikit-

learn.org/stable/modules/generated/sklearn.svm.SVR.html. [Accessed: 22-Feb-2018].
[67] L. Garcia, M. Vatn, F. Mühle, and L. Sætran, “Experiments in the wind turbine far wake for the evaluation of an analytical wake model,” *J. Phys. Conf. Ser.*, vol. 854, no. 1, 2017.
[68] Z. Shao, Y. Wu, L. Li, S. Han, and Y. Liu, “Multiple Wind Turbine Wakes Modeling Considering the Faster Wake Recovery in Overlapped Wakes,” *Energies*, vol. 12, no. 4, p. 680, Feb. 2019.
[69] M. Gaumond, P.-E. Réthoré, S. Ott, A. Peña, A. Bechmann, and K. S. Hansen, “Evaluation of the wind direction uncertainty and its impact on wake modeling at the Horns Rev offshore wind farm,” *Wind Energy*, vol. 17, no. 8, pp. 1169–1178, Aug. 2014.
[70] I. Katic, J. Højstrup, and N. O. Jensen, “A Simple Model for Cluster Efficiency,” in *EWEC’86. Proceedings*, 1986, vol. 1.
[71] T. Göçmen, P. Van Der Laan, P. E. Réthoré, A. P. Diaz, G. C. Larsen, and S. Ott, “Wind turbine wake models developed at the technical university of Denmark: A review,” *Renew. Sustain. Energy Rev.*, vol. 60, pp. 752–769, 2016.