Renewable Energy xxx (xxxx) xxx

Contents lists available at ScienceDirect

Renewable Energy

journal homepage: www.elsevier.com/locate/renene

Probabilistic modelling of wind turbine power curves with application of heteroscedastic Gaussian Process regression

T.J. Rogers [a, *], P. Gardner [a], N. Dervilis [a], K. Worden [a], A.E. Maguire [b], E. Papatheou [c], E.J. Cross [a]

[a] Dynamics Research Group, Department of Mechanical Engineering, University of Sheffield, Mappin Street, Sheffield, S1 3JD, UK, England
[b] Vattenfall Wind Power, Holyrood Road, Edinburgh, Scotland, EH8 8PJ, UK
[c] College of Engineering Mathematics and Physical Sciences, University of Exeter, N Park Road, Exeter, EX4 4QF, UK

ARTICLE INFO

Article history:
Received 1 May 2019
Received in revised form
12 September 2019
Accepted 30 September 2019
Available online xxx

Keywords:
Wind turbine
Power curve
Gaussian process
Heteroscedastic
Probabilistic
Bayesian

ABSTRACT

There exists continued interest in building accurate models of wind turbine power curves for better understanding of performance or assessment of the condition of the turbine or both. Better predictions of the power curve allow increased insight into the operation of the turbine, aid operational decision making, and can be a key feature of online monitoring and fault detection strategies. This work proposes the use of a heteroscedastic Gaussian Process model for this task. The model has a number of attractive properties when modelling power curves. These include, removing the need to specify a parametric functional form for the power curve and automatic quantification of the variance in the prediction. The model exists within a Bayesian framework which exhibits built-in protection against over-fitting and robustness to noisy measurements. The model is shown to be effective on data collected from an operational wind turbine, returning accurate mean predictions (< 1% normalised mean-squared error) and higher likelihoods than a corresponding homoscedastic model.

1. Introduction

The power curve of a wind turbine is one of its key performance indicators. As the popularity of wind-based power generation continues to grow, the characterisation of the performance of turbines is an important step in the justification for, and management of, this renewable energy source. Being able to accurately predict the power output of a turbine has a number of beneficial use cases for the operator. The prediction of power output allows more accurate prediction of expected income from the turbine (and by extension, farm) allowing for more forward-thinking business planning. Alternatively, the power curve of the turbine has been shown to be an effective indicator of degradation in performance of the system, for example see Papatheou et al. [1]. That work sits within a wider body of work on monitoring wind turbines, usually via SCADA (supervisor control and data acquisition) systems in order to infer the structural condition of the turbine — this being one example of structural health monitoring [2].

Typical power curve data collected from a wind turbine SCADA system is shown in Fig. 1. Visually, it can be seen that the power curve exhibits a number of interesting features from the modelling perspective. The relationship between the wind speed and the power output is nonlinear. The data has a stochastic element or there is noise in the measurement of the data and that this noise is not constant across the input domain. This input-dependent noise variance is referred to as heteroscedasticity, as opposed to homoscedastic noise where the variance of the noise is independent of the input. Finally, there are a number of data points which could be considered outlying from the bulk of the data distribution. The combination of these factors makes modelling the behaviour and variance of the power curve robustly an interesting and challenging prospect.

This paper presents a methodology for building probabilistic models of wind turbine power curves based on a heteroscedastic Gaussian Process method. This allows predictions to be made of the mean and variance which approximate the distribution of power output from a turbine given the measured wind speed. The question remains: why might this be useful to the end user?

Considering applications in Structural Health Monitoring (SHM) [2,3], the value of a probabilistic model is made apparent. Here,

* Corresponding author.
E-mail address: tim.rogers@sheffield.ac.uk (T.J. Rogers).

https://doi.org/10.1016/j.renene.2019.09.145

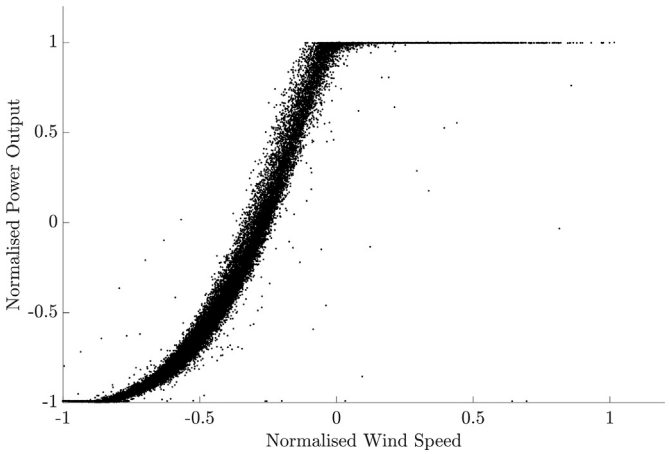

Fig. 1. Typical power curve data from a wind turbine SCADA sytem that has been normalised for anonymity.

probabilistic methods provide a natural framework in which decisions can be made based on quantifiable risk. Most simply, there is a distinction between saying there is or is not damage, as compared to considering the probability of damage on a structure. Alternatively, if predicting fatigue damage accrual, providing a distribution over expected crack lengths from a model returns information, not only about the most likely result, but also confers a measure of confidence which can guide the decision-making process. Finally, it is worth considering that a deterministic approach to these problems does not remove the inherent uncertainty from the process but it does fail to account for it. By failing to acknowledge uncertainty when attempting to understand a structure's condition, the engineer implies perfection in models and processes which are inherently imperfect. In safety critical applications, this can lead to failure in planning for unlikely events, greatly magnifying the consequences of these.

Beyond the realm of SHM, the quantification of uncertainty continues to add value. For example, one key use of power curve models is for investors to calculate expected returns from a turbine or farm. Models which can account not only for the mean trend but also quantify uncertainty in predictions allow financial planning to be done on the basis of more information. Clearly there exists a distribution over wind speeds that a turbine will be subject to. An uncertain model of the power output of the turbine allows combination of these distributions. The possession of models which quantify and handle uncertainty allows for robust *uncertainty propagation*. In this, all the uncertainty present throughout the power generation process can be combined to give a distribution over a variable of interest — e.g. monthly income. Again, being in possession of this distribution, allows better confidence in the models and enables long-term risk-based financial planning.

It is hopefully clear that the accurate modelling of uncertainty offers tangible cost benefit across a range of situations. This includes day to day benefits in operation such as health monitoring applications or longer term benefits in assistance with high-level financial planning. The final reason for building probabilistic models of uncertainty, for systems such as wind turbines, is that it is possible. To not do so fails to make full use of the data which has been collected. The process of sensing and data acquisition remains expensive and difficult in comparison to building and learning data-based models. By reducing this data to a single deterministic line, users fail to make full use of this valuable resource. If operators are willing to spend money to acquire data, it is only sensible to build the most expressive model possible.

1.1. Related work

The task of modelling wind turbine power curves has been explored in the literature previously with review papers being published in 2011 [4], 2013 [5], and 2014 [6]. Broadly speaking, approaches to solving this task have been separated into those which aim to build models based on physical/engineering understanding of the behaviour of the turbine and those which rely solely on learning from data.

A number of models have been built upon polynomial regression equations; the most common being those based on the cubic relationship between wind speed and maximum available power, e.g. Carrillo et al. [5]. However, attempts have been made to fit higher-order polynomial models, a 6^{th} order in Ref. [7] and a 9^{th} order in Ref. [8]. However, these works routinely fail to use any form of regularisation or cross validation, which is required to ensure that the models will generalise and fit well to unseen data. Marčiukaitis et al. [7] discuss the use of a cross-validation technique for model assessment; while this demonstrates the consistency of the model, it does not provide any protection against over-fitting during the training stage. The problem of over-fitting occurs most frequently in over-parameterised models, the classic examples being high-order polynomials; further discussion of this problem and techniques to alleviate it can be found in (for example) Bishop [9] or Barber [10]. Taslimi-Renani et al. [11] also discuss the problem of overfitting in their work where a modified hyperbolic tangent function is proposed as a parametric model of the power curve. In these references and in this work, distinction is made between two subsets of data; training data which is used for learning the model, and (independent/unseen) testing data which is unused in learning the model but is used to assess the expected performance of the model in operation. Results are presented on both the training data and this unseen test data to demonstrate the ability of the model to generalise (i.e. continue to make valid predictions for the turbine into the future).

Other parametric methods have explored fitting functions which, heuristically, match the shape of the power curve. These have included variations on logistic and hyperbolic tangent functions, for example see Lydia et al. [6], Marčiukaitis et al. [7], Seo et al. [12]. In a similar manner Villanueva and Feijóo [13] and Lydia et al. [14] propose parametric models of the power curve based upon a logistic function. These functions have the benefit of possessing many of the properties that appear inherent to the data in a wind turbine power curve, i.e. boundedness at high and low wind speeds and nonlinear transition between these bounds.

For modelling power curves, the use of (artificial) neural networks has been explored e.g. Refs. [15,16] and more recently this trend has continued [17,18] in line with the continued popularity of neural networks across many fields. The use of a support vector machine (SVM) was also discussed in Ouyang et al. [19], where the SVM is created based on using the centroids of a k-means algorithm as training data. Yan et al. [20] consider the combination of a number of deterministic models with approaches that also attempt to capture the uncertainty in the power curve, a comparison is made between these approaches in terms of the error and an "expectation variance ratio". Wang et al. [21] propose a probabilistic approach to modelling the wind turbine power curve based spline regression models which are used to generate inputs to a neural network for power forecasting. The use of Gaussian Process (GP) regression models for modelling the wind turbine power curve has, also, previously been discussed. In Papatheou et al. [1], Antoniadou et al. [22] and Papatheou et al. [23] the use of the standard GP formulation allows detection of damage in the turbine. In Manobel et al. [24] the GP is used as a pre-processing step for filtering data before it is passed to a neural network model. This

Please cite this article as: T.J. Rogers et al., Probabilistic modelling of wind turbine power curves with application of heteroscedastic Gaussian Process regression, Renewable Energy, https://doi.org/10.1016/j.renene.2019.09.145

work appears to overcomplicate the problem of modelling the power curve. The addition of the neural network seems superfluous since there exist proofs that the GP is a universal approximator [25]. In addition to this, it can be shown that, as the number of neurons in a single layer MLP (multi-layer perceptron) tends to infinity, a GP is recovered [26] — the extension of this work to deeper networks is discussed in Ref. [27]. Additionally, the use of the GP for filtering forces the data into an approximately Gaussian distribution, which is homoscedastic. This leads to the exclusion of potentially valid points as outliers and can distort the true distribution of the data. Pandit et al. [28] consider the use of a standard GP where the air density is included as a second input variable along with the wind speed, this shows improvement in accuracy. In Pandit et al. [29] a model similar to this one is compared to a support vector machine and a random forest model, results reveal the GP to score better in the performance metrics shown. It can be seen that there have been numerous approaches to modelling the wind turbine power curve and that investigation into this problem continues to be an active research area. However, a number of the models used make no attempt to model the uncertainty in the power output of the turbine and of those that do a heteroscedastic approach is very rarely taken. The work contained in this paper aims to provide a methodology that forms and an accurate model of the power output of the wind turbine while also quantifying this varying uncertainty across different wind speeds in a manner which is efficient for large datasets and statistically rigorous.

The layout of the paper is as follows; in section 2 the necessary theory for Gaussian Process regression is introduced, section 2.1 discusses the efficient modelling of large datasets with GPs; section 2.2 extends the GP model to the heteroscedastic noise case; section 2.3 combines these approaches to form a sparse heteroscedastic model, and finally section 2.4 presents a methodology for distributed computation of these models and combination via a robust Bayesian committee machine. The use of the model is shown in section 3 where it is applied to data measured from an operational wind turbine. The benefit of moving to a heteroscedastic model is demonstrated by comparison with a homoscedastic GP model, both quantitatively and qualitatively. Finally, conclusions are made in section 4 with discussion of possible directions for future work.

2. Gaussian Process regression

Gaussian Process (GP) models provide a flexible Bayesian machine learning method for solving regression problems [10,30–32]. They exhibit a number of desirable properties for this application: they are nonparametric, automatically quantify uncertainty in predictions, require little *a priori* input, and are capable of modelling signals even in the presence of high noise levels on the measured data. The GP allows a prior distribution to be placed over an entire function for inference rather than merely learning the parameters of a model. The GP is developed for modelling functions of the form,

$$y = f(\mathbf{x}) + \varepsilon \qquad \varepsilon \sim \mathcal{N}\left(0,\ \sigma_n^2\right) \tag{1}$$

i.e. it models data as the output of some function $f(\mathbf{x})$, operating on a D-dimensional input $\mathbf{x}$. This function is corrupted by some additive Gaussian noise ε with zero mean and a fixed variance σ_n^2.

The most common — and most intuitive — introduction to the GP is as a distribution over functions, where a single draw from the GP is a potential realisation of a function generated by that GP. In this way the GP can be seen as the prior over $f(\mathbf{x})$ in equation (1). A GP is defined as in equation (2), where $\mathbf{x}$ and $\mathbf{x}$' are a pair of inputs to the function of interest,

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \tag{2}$$

It follows that the GP is characterised completely by its mean, $m(\mathbf{x})$, and covariance, $k(\mathbf{x}, \mathbf{x}')$, functions. To make predictions, the joint Gaussian distribution between the training and testing data is assessed,

$$\begin{bmatrix} \mathbf{y} \\ y_\star \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m(X) \\ m(\mathbf{x}_\star) \end{bmatrix}, \begin{bmatrix} K_{XX} + \sigma_n^2 \mathbb{I} & K_{X\mathbf{x}_\star} \\ K_{\mathbf{x}_\star X} & K_{\mathbf{x}_\star\mathbf{x}_\star} + \sigma_n^2 \mathbb{I} \end{bmatrix} \right) \tag{3}$$

Here, the notation X is used to denote a set of N, D dimensional, training inputs where $X \in \mathbb{R}^{N\times D}$ and $\mathbf{y}$ is the corresponding set of N measured training outputs with $\mathbf{y} \in \mathbb{R}^{N\times 1}$. When wanting to predict with the model, a new input $\mathbf{x}_\star$ can be considered (trivially this could also be $X_\star$ if predicting at multiple points). This is used to make a prediction at a new potentially unknown output $y_\star$. By the properties of a multivariate Gaussian, every conditional distribution is also Gaussian. Using this standard result, it is possible to write down the predictive distributions over $y_\star$,

$$p(y_\star | \mathbf{x}_\star, X, \mathrm{y}) = \mathcal{N}(\mathbb{E}[y_\star], \mathbb{V}[y_\star])$$

$$\mathbb{E}[y_\star] = m(\mathbf{x}_\star) + K_{\mathbf{x}_\star X}\left(K_{XX} + \sigma_n^2 \mathbb{I}\right)^{-1}(\mathbf{y} - m(X))$$

$$\mathbb{V}[y_\star] = K_{\mathbf{x}_\star\mathbf{x}_\star} - K_{\mathbf{x}_\star X}\left(K_{XX} + \sigma_n^2 \mathbb{I}\right)^{-1} K_{X\mathbf{x}_\star} + \sigma_n^2 \tag{4}$$

It is possible to assess new test points since the covariance of the process is fully described by the covariance function. This together with the mean function $m(\cdot)$ allows the GP to be used when predicting at any new $\mathbf{x}_\star$. The mean function can be chosen to be any parametric function of the inputs, although it is commonly set to zero when the GP is presented in machine learning literature [31].

In order to fully specify the GP model, a covariance (kernel) function must be chosen which defines the similarity of any two sets of input points giving rise to the covariance matrix K. A popular choice for the covariance function is the squared-exponential (SE), which is defined for two input points $\mathbf{x}$ and $\mathbf{x}$' as,

$$k_{\mathrm{SE}}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left\{\frac{||\mathbf{x} - \mathbf{x}'||^2}{2l}\right\} \tag{5}$$

The use of the squared-exponential kernel embeds the belief that the function being modelled is infinitely differentiable.[1] It should be noted that by choosing this covariance function the user is restricting the functions which can be modelled to those which conform to these properties.

It can be seen that there exist a small number of *hyperparameters* in the kernel which must be determined in order to make use of the GP. In the case of the squared-exponential covariance these are the signal variance σ_f^2 and the length scale ℓ. These two hyperparameters control the behaviour of the covariance function; σ_f^2 can be interpreted as the prior variance of the signal being modelled and ℓ mediates the region of influence of the kernel. In other words, the length-scale controls how smooth the function being modelled is, where increasing the length-scale increases the smoothness of the function.

[1] Formally this property is referred to as smoothness, however, to avoid confusion the term smooth is not used in this paper. Instead the word smooth is used to refer to functions which vary more slowly with relation to the input space, i.e. have longer length-scales.

Please cite this article as: T.J. Rogers et al., Probabilistic modelling of wind turbine power curves with application of heteroscedastic Gaussian Process regression, Renewable Energy, https://doi.org/10.1016/j.renene.2019.09.145

In order to learn these hyperparameters, a Type-II maximum likelihood approach is taken as in Ref. [31]. The *marginal likelihood* of the model, also referred to as the model evidence, is maximised. This optimisation makes use of the Bayesian Occam's razor [33–35] to find the minimally complex model given the observed data in the training set $\mathscr{D} = \{X, \mathrm{y}\}$. This optimisation is normally performed as a minimisation over the negative log marginal likelihood for convenience and numerical stability. Thus, an estimate of the hyperparameters $\widehat{\boldsymbol{\theta}} = \{\sigma_f^2, \ell\}$ is obtained through the following optimisation,

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \{ -\log p(\mathbf{y} \mid X, \boldsymbol{\theta}) \} \tag{6}$$

with,

$$\begin{aligned} -\log p(\mathbf{y}|X,\boldsymbol{\theta}) &= -\log \mathscr{N}\left(\mathbf{y}\middle|m(X), K_{XX}+\sigma_n^2\mathbb{I}\right) \\ &= \frac{N}{2}\log(2\pi) + \\ &\frac{1}{2}\log\left|K_{XX}+\sigma_n^2\mathbb{I}\right| + \\ &\frac{1}{2}\left[\left(\mathbf{y}-m\left(X\right)\right)^T\left(K_{XX}+\sigma_n^2\mathbb{I}\right)^{-1}\left(\mathbf{y}-m\left(X\right)\right)\right] \end{aligned} \tag{7}$$

In this way the hyperparameters of the kernel (and if necessary the parameters of the mean function) can be learnt and the GP is fully specified by equations (4) and (7).

2.1. Handling large datasets

In order to either learn the hyperparameters of the GP or to make predictions, it is necessary to assess the inverse of the covariance matrix with noise, $(K_{XX}+\sigma_n^2\mathbb{I})^{-1}$. This operation is $\mathscr{O}(N^3)$ in both computation and memory storage. Practically, this means that for datasets larger than roughly ten thousand data points it is not feasible to learn a GP model. This is the case in many datasets collected from SCADA systems where the number of datapoints regularly exceeds tens, if not hundreds, of thousands. A number of methods have been considered to address this problem. This class of models is referred to as *sparse Gaussian Processes*. The most common methodology is to introduce a number of *inducing points*, although other methods have been explored [36,37]. Introducing these inducing points reduces the complexity of the process from $\mathscr{O}(N^3)$, for N datapoints, to $\mathscr{O}(NM^2)$, for M inducing points, giving advantage when $M \ll N$. Broadly speaking, inducing point methods can be separated into two classes, *model approximations* and *posterior approximations*. Model approximations modify the prior of the model to achieve sparsity whereas posterior approximations approximate the posterior directly. In Quiñonero-Candela and Rasmussen [38], or more recently in Bui et al. [39], the use of inducing points (also referred to as pseudo-points) is brought under unifying frameworks. For the sake of brevity, the content of these papers is not duplicated here.

In general, posterior approximations of the GP will result in more robust models than model approximations [40]. It is known that a posterior approximation is not able to overfit the data unlike, for instance, a Fully Independent Training Conditional (FITC) approach [40,41]. Therefore, a posterior approximation approach is adopted in this work, namely the Variational Free Energy (VFE) approach [42].

A very brief review of the VFE sparse GP is given here, for a full introduction the reader is referred to Ref. [42]. The inducing points of the model $\{Z, \mathbf{u}\}$ (where Z contains the locations of the inducing points and $\mathbf{u}$ the values of the latent function at those points) are used to form a *variational approximation* of the full posterior of the model. The model can then be learnt by minimising the Kullback-Leibler (KL) divergence between this approximate joint posterior and the full joint GP posterior. The joint variational (approximate) posterior $q(\mathbf{f}, \mathbf{u})$ is formed such that,

$$q(\mathbf{f}, \mathbf{u}) = p(\mathbf{f}, \mathbf{u})\varphi(\mathbf{u}) \tag{8}$$

where $\varphi(\mathbf{u})$ is known as the 'free' variational distribution with $\mathbf{f}$ being the latent function values at the measured inputs and $\mathbf{u}$ dependent upon the set of 'free' inputs Z. This allows the joint posterior of the GP $p(\mathbf{f}, \mathbf{f}_\star)$ to be approximated directly as,

$$p(\mathbf{f}, \mathbf{f}_\star) \approx q(\mathbf{f}, \mathbf{f}_\star) = \int p(\mathbf{f}_\star|\mathbf{u}) q(\mathbf{f}|\mathbf{u}) p(\mathbf{u}) \mathrm{d}\mathbf{u} \tag{9}$$

It is possible to find the optimal choice of $\varphi(\mathbf{u})$ analytically from which a lower bound on the marginal likelihood, $F(Z)$, can be established as,

$$\begin{aligned} FZ = &-\frac{1}{2}\log\left|Q_{XX}+\sigma_n^2\mathbb{I}\right| - \frac{1}{2}(\mathbf{y}-m(X))^T\left(Q_{XX}+\sigma_n^2\mathbb{I}\right)^{-1} \\ &\times (\mathbf{y}-m(X)) - \frac{N}{2}\log 2\pi - \frac{1}{2\sigma_n^2}\mathrm{tr}\left(K_{XX}-Q_{XX}\right) \end{aligned} \tag{10}$$

where, $\mathrm{tr}(\cdot)$ is the trace operator and the approximate covariance Q_{XX} is defined as,[2]

$$Q_{XX} = K_{X\mathbf{u}}K_{\mathbf{uu}}^{-1}K_{\mathbf{u}X} \tag{11}$$

Now in learning the hyperparameters of the GP the bound in equation (10) is used in place of the marginal likelihood $p(\mathbf{y}|X, \boldsymbol{\theta})$ in equation (6). Predictions can then be made through this approximate posterior in a similar manner to the standard GP. The predictive distribution of the VFE model is given by,

$$\begin{aligned} q(y_\star|\mathbf{x}_\star, X, \mathbf{y}, \mathbf{u}) &= \mathscr{N}(\mathbb{E}[y_\star], \mathbb{V}[y_\star]) \\ \mathbb{E}[y_\star] &= Q_{\mathbf{x}_\star X}\left(Q_{XX}+\sigma_n^2\mathbb{I}\right)^{-1}\mathbf{y} \\ \mathbb{V}[y_\star] &= K_{\mathbf{x}_\star\mathbf{x}_\star} - Q_{\mathbf{x}_\star X}\left(Q_{XX}+\sigma_n^2\mathbb{I}\right)^{-1}Q_{X\mathbf{x}_\star} \end{aligned} \tag{12}$$

By making use of this sparse approximation, the computational requirements for a dataset with N datapoints is reduced from $\mathscr{O}(N^3)$ to $\mathscr{O}(NM^2)$ for M inducing points. This now makes it feasible to handle large engineering datasets such as those returned by a data acquisition system installed on a wind turbine.

2.2. Heteroscedastic noise models

Considering the data shown in Fig. 1, it can be seen that one of the key assumptions in the GP does not hold when modelling power curve data. That is the assumption of *homoscedastic* noise, this is that the noise on the function $f(\mathbf{x})$ is an additive Gaussian noise with fixed variance. In fact, it can be seen that the noise variance changes across the input space, i.e. with changing wind speed there is a change in noise variance. In a *heteroscedastic* noise model it is assumed that the noise model is a function of the inputs to the system. The regression model introduced in equation (1) would

[2] Here notation is established for a general matrix Q_{ab} such that $Q_{\mathbf{ab}} = K_{\mathbf{au}}K_{\mathbf{uu}}^{-1}K_{\mathbf{ub}}$.

Please cite this article as: T.J. Rogers et al., Probabilistic modelling of wind turbine power curves with application of heteroscedastic Gaussian Process regression, Renewable Energy, https://doi.org/10.1016/j.renene.2019.09.145

then become,

$$y = f(\mathbf{x}) + \varepsilon(\mathbf{x}) \qquad \varepsilon \sim \mathcal{N}(0,\ r(\mathbf{x})) \tag{13}$$

It can be seen that the variance of the noise process is now considered to be a function of the inputs to the model. In the case of the power curve, this expresses the fact that noise variance is dependent on wind speed.

In the same way that a GP prior can be used to infer the unknown function $f(\mathbf{x})$, such that $f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}),\ k_f(\mathbf{x},\ \mathbf{x}'))$. It is possible to model the function over the noise variance using a GP. This was first presented in Lázaro-Gredilla and Titsias [43], where, since the variance of the noise is strictly positive the function $r(\mathbf{x})$ is modelled as the exponential of a Gaussian Process regression. That is,

$$r(\mathbf{x}) = \exp\{g(\mathbf{x})\} \tag{14}$$

where,

$$g(\mathbf{x}) \sim \mathcal{GP}(\mu_0, k_g(\mathbf{x}, \mathbf{x}')) \tag{15}$$

The GP which is used to model $g(\mathbf{x})$ is assigned its own covariance function $k_g(\mathbf{x}, \mathbf{x}')$ and is considered to have a constant mean μ_0. The addition of the second GP over the log noise variance increases the expressive power of the model, but with that, the difficultly in learning and inference. The second GP increases the number of hyperparameters that must be learnt by the number required to express the constant mean and kernel of the second GP.

The introduction of this heteroscedastic noise model also means that the marginal likelihood and predictive equations of the model are no longer available in closed form. To handle this, a variational approximation is used. Similarly to the VFE sparse method the approximate distribution over the posterior is used to form a lower bound on the marginal likelihood of the model which can be found to be dependent on two sets of parameters $\boldsymbol{\mu}$ and Σ. The bound is found in Ref. [43] to be,

$$F(\boldsymbol{\mu}, \Sigma) = \log \mathcal{N}\left(\mathbf{y}|\mathbf{0}, K_f + R\right) - \frac{1}{4}\mathrm{tr}(\Sigma) - \mathrm{KL}\big(\mathcal{N}(\mathbf{g}|\boldsymbol{\mu}, \Sigma) \times ||\mathcal{N}(\mathbf{g}|\mu_0\mathbf{1}, K_g)\big) \tag{16}$$

Here, K_f and K_g are used to denote the covariance matrices of the two Gaussian Processes over $f(\mathbf{x})$ and $g(\mathbf{x})$ respectively. $\mathrm{KL}(p(a)||p(b))$ is the Kullback-Leibler divergence between distribution $p(a)$ *pofa* and $p(b)$; $\mathbf{1}$ is a vector of ones; $\boldsymbol{\mu}$ and Σ are variational parameters to be determined, and R is a diagonal matrix whose diagonal elements are given by,

$$R_{ii} = \exp\left\{\mu_i - \frac{1}{2}\Sigma_{ii}\right\} \quad i = 1, \ldots, N \tag{17}$$

It can be seen that, in $\boldsymbol{\mu}$ and Σ, there exist $N + N(N+1)/\ 2$ unknown free variational parameters which must be learnt. Following the approach in Lázaro-Gredilla and Titsias [43], it is possible to reparameterise $\boldsymbol{\mu}$ and Σ in terms of Λ — a diagonal semi-positive-definite matrix — reducing the number of parameters to be learnt to N. This allows $\boldsymbol{\mu}$ and Σ to be expressed in the following form,

$$\boldsymbol{\mu} = K_g\left(\Lambda - \frac{1}{2}\mathbb{I}\right)\mathbf{1} + \mu_0\mathbf{1}, \qquad \Sigma^{-1} = K_g^{-1} + \Lambda \tag{18}$$

This being the case, the bound on the marginal likelihood can be computed and the hyperparameters of the model can be learnt. The overall increase in computational load for the heteroscedastic GP model means that learning takes roughly twice as long as a homoscedastic GP [43].

One final complication in the heteroscedastic GP model is that the full predictive distribution is also unavailable in closed form. To obtain it would require evaluating the integral,

$$q(y_\star) = \iint p(y_\star|g_\star, f_\star) q(f_\star) q(g_\star) \mathrm{d}f_\star \mathrm{d}g_\star = \int \mathcal{N}\left(y_\star \middle| a_\star, c_\star^2 + \exp\{g_\star\}\right) \mathcal{N}\left(g_\star \middle| \mu_\star, \sigma_\star^2\right) \mathrm{d}g_\star \tag{19}$$

where,

$$a_\star = k_f(\mathbf{x}_\star, X)\left(K_f + R\right)^{-1}\mathbf{y} \tag{20a}$$

$$c_\star^2 = k_f(\mathbf{x}_\star, \mathbf{x}_\star) - k_f(\mathbf{x}_\star, X)\left(K_f + R\right)^{-1} k_f(X, \mathbf{x}_\star) \tag{20b}$$

$$\mu_\star = k_g(\mathbf{x}_\star, X)\left(\Lambda - \frac{1}{2}\mathbb{I}\right)\mathbf{1} + \mu_0 \tag{20c}$$

$$\sigma_\star^2 = k_g(\mathbf{x}_\star, \mathbf{x}_\star) - k_g(\mathbf{x}_\star, X)\left(K_g + \Lambda^{-1}\right)^{-1} k_g(X, \mathbf{x}_\star) \tag{20d}$$

Although equation (19) cannot be computed in closed form, it is possible to calculate the first two moments of the predictive distribution $q(y_\star)$; that is, the mean and the variance of this distribution.[3] These moments can be found to be,

$$\mathbb{E}_{q(y_\star)}[y_\star] = a_\star \tag{21a}$$

$$\mathbb{V}_{q(y_\star)}[y_\star] = c_\star^2 + \exp\left\{\mu_\star + \frac{1}{2}\sigma_\star^2\right\} \tag{21b}$$

Therefore, it is possible to make predictions using a GP under a heteroscedastic noise assumption. By calculating only the first two moments of the predictive distribution, the distribution over an unknown output $y_\star$ given a test input $\mathbf{x}_\star$ can be approximated. An approximation of this distribution by its first two moments is to assume that the distribution over the test output at this point is well represented by its first two moments; the true distribution may not be Gaussian but by using only the first two moments is assumed to be close to this. This allows a probabilistic prediction of the function of interest to be made while also predicting the variance of the function at any given input. This additional information regarding the uncertainty on the process is invaluable if the predictions are to be carried forward into further analysis.

2.3. Sparse heteroscedastic Gaussian Process regression

In possession of both a sparse and a heteroscedastic Gaussian Process model it is natural to explore the combination of these into a sparse heteroscedastic GP. This combination has also been shown in the literature by Liu et al. [44]. In that work, the authors establish a new variational bound when making a variational approximation of the posterior under both the heteroscedastic model and a sparse model akin to the VFE approach, which they term the Variational Sparse Heteroscedastic Gaussian Process (VSHGP). Taking the heteroscedastic model shown in Ref. [43] and presented in section 2.2, it is possible to form a sparse heteroscedastic GP. Separate sets of inducing points are introduced to both the function GP modelling $f(\mathbf{x})$ and the log noise variance GP modelling $g(\mathbf{x})$. The same

[3] Please note, the explicit conditioning of the posteriors on the training data, test input, inducing points, and hyperparameters is dropped for simplicity of notation.

Please cite this article as: T.J. Rogers et al., Probabilistic modelling of wind turbine power curves with application of heteroscedastic Gaussian Process regression, Renewable Energy, https://doi.org/10.1016/j.renene.2019.09.145

strategy for achieving sparsity is followed as in section 2.1 (using the technique of Titsias [42]).

The variational approximation of this complete model again reduces to determining a lower bound on the marginal likelihood. This is found to be,

$$\begin{aligned} F(\boldsymbol{\mu},\Sigma) = \log \mathcal{N}\left(\mathbf{y}\middle|\mathbf{0},\ Q_{XX}^{(f)}+R\right) \\ -\frac{1}{4}\mathrm{tr}\left(\Sigma_g\right) \\ -\frac{1}{2}\mathrm{tr}\left(R^{-1}\left(K_{XX}^{(f)}-Q_{XX}^{(f)}\right)\right) \\ -\mathrm{KL}\left(\mathcal{N}(\mathbf{g}|\boldsymbol{\mu}_{\mathbf{u}},\Sigma_u)\middle|\middle|\mathcal{N}(\mathbf{g}|\mu_0,K_{uu}^{(g)})\right) \end{aligned} \tag{22}$$

where,

$$\boldsymbol{\mu}_{\mathbf{g}} = \Omega_{Xu}^{(g)}\left(\boldsymbol{\mu}_{\mathbf{u}}-\mu_0\mathbf{1}\right)+\mu_0\mathbf{1} \tag{23a}$$

$$\Sigma_g = K_{XX}^{(g)} - Q_{XX}^{(g)} + \Omega_{Xu}^{(g)}\,\Sigma_u\,\Omega_{uX}^{(g)} \tag{23b}$$

$$\boldsymbol{\mu}_{\mathbf{u}} = K_{uX}^{(g)}\left(\Lambda-\frac{1}{2}\mathbb{I}\right)\mathbf{1}+\mu_0\mathbf{1} \tag{23c}$$

$$\Sigma_u^{-1} = \left[K_{uu}^{(g)}\right]^{-1}+\Omega_{uX}^{(g)}\Lambda\Omega_{Xu}^{(g)} \tag{23d}$$

given,

$$\Omega_{Xu}^{(g)} = K_{Xu}^{(g)}\left[K_{uu}^{(g)}\right]^{-1}\Omega_{uX}^{(g)} = \left[K_{uu}^{(g)}\right]^{-1}K_{uX}^{(g)} \tag{24}$$

and R is a diagonal matrix with elements,

$$R_{ii} = \exp\left\{\left[\mu_g\right]_i-\frac{1}{2}\left[\Sigma_g\right]_{ii}\right\} \tag{25}$$

At this point it is worth clarifying the notation used in this section. The introduction of a sparse approximation to the two Gaussian Processes in the heteroscedastic model adds an additional number of hyperparameters corresponding to the inducing points used in $f(\mathbf{x})$ and $g(\mathbf{x})$. It should be noted that the inducing points for $f(\mathbf{x})$ and $g(\mathbf{x})$ are two separate sets that can be different sizes. In light of this, notationally a covariance matrix is indexed by a superscript (f) or (g) to denote which function — and therefore hyperparameters — are being considered. The subscript is used to denote which sets of points the covariance is taken between, with X being the full measured set of inputs and u being the set of inducing points for that function. For example, $K_{uX}^{(g)}$ indicates the matrix of covariances between the inducing points of the process for $g(\mathbf{x})$ and the training data X given those learnt inducing points and the hyperparameters of the kernel for the log noise process. Although there is a non-trivial amount of algebra to arrive at this point, the bound developed in equation (22) can be used in place of the marginal likelihood of the standard GP $p(\mathbf{y}|X,\boldsymbol{\theta})$ to learn the set of hyperparameters of the model $\boldsymbol{\theta}$. However, the number of hyperparameters which must be learnt has now increased to include the hyperparameters for the kernels $k_f(\mathbf{x},\mathbf{x}')$ and $k_g(\mathbf{x},\mathbf{x}')$, the constant mean for the log noise variance μ_0, the set of M_f inducing points for $f(\mathbf{x})$, the set of M_g inducing points for $g(\mathbf{x})$, and the N variational parameters which form the diagonal matrix Λ.

Turning attention to making predictions with the VSHGP model, it is necessary to compute the approximate posterior distribution over $y_\star$ — $q(y_\star)$. As with the non-sparse heteroscedastic GP the computation of this approximate posterior requires the computation of an intractable integral,

$$\begin{aligned} q(y_\star) &= \iint p(y_\star|g_\star,f_\star)q(f_\star)q(g_\star)\mathrm{d}f_\star\mathrm{d}g_\star \\ &= \int \mathcal{N}\left(y_\star\middle|\mu_\star^{(f)},\exp\{g_\star\}+\sigma_\star^{2^{(f)}}\right)\mathcal{N}\left(g_\star\middle|\mu_\star^{(g)},\sigma_\star^{2^{(g)}}\right)\mathrm{d}g_\star \end{aligned} \tag{26}$$

This equation is dependent upon a predictive mean and variance for $f(\mathrm{x})$ ($\mu_\star^{(f)}$ and $\sigma_\star^{2^{(f)}}$) as well as the predictive mean and variance for $g(\mathrm{x})$ ($\mu_\star^{(g)}$ and $\sigma_\star^{2^{(g)}}$). Each of these can be computed in closed form [44], defining K_R as,

$$K_R = K_{uX}^{(f)}R^{-1}K_{Xu}^{(f)}+K_{uu}^{(f)} \tag{27}$$

$$\mu_\star^{(f)} = K_{\star u}^{(f)}K_R^{-1}K_{uX}^{(f)}R^{-1}\mathbf{y} \tag{28a}$$

$$\sigma_\star^{2^{(f)}} = K_{\star\star}^{(f)}-K_{\star u}^{(f)}\left[K_{uu}^{(f)}\right]^{-1}K_{u\star}^{(f)}+K_{\star u}^{(f)}K_R^{-1}K_{u\star}^{(f)} \tag{28b}$$

$$\mu_\star^{(g)} = K_{\star u}^{(g)}\left[K_{uu}^{(g)}\right]^{-1}\left(\boldsymbol{\mu}_{\mathbf{u}}-\mu_0\mathbf{1}\right)+\mu_0\mathbf{1} \tag{28c}$$

$$\sigma_\star^{2^{(g)}} = K_{\star\star}^{(g)}-K_{\star u}^{(g)}\left[K_{uu}^{(g)}\right]^{-1}K_{u\star}^{(g)}+K_{\star u}^{(g)}\left[K_{uX}^{(g)}\Lambda^{-1}K_{Xu}^{(g)}+K_{uu}^{(g)}\right]^{-1}K_{u\star}^{(g)} \tag{28d}$$

In an analogous manner to (21), the first two moments of $q(y_\star)$ under the VSHGP model can we written down as,

$$\mathbb{E}_{q(y_\star)}[y_\star] = \mu_\star^{(f)} \tag{29a}$$

$$\mathbb{V}_{q(y_\star)}[y_\star] = \sigma_\star^{2^{(f)}}+\exp\left\{\mu_\star^{(g)}+\frac{1}{2}\sigma_\star^{2^{(g)}}\right\} \tag{29b}$$

These first two moments can then be assumed to well represent the full predictive distribution, i.e. it is assumed that this distribution is approximately Gaussian.

2.4. Distributed computation

In Ref. [44] one further extension is made to this model. The Distributed Variational Sparse Heteroscedastic Gaussian Process, is presented where the data are divided into a number of subsets. Each of these subsets is learnt via a separate VSHGP in the manner described above using the bound established in equation (22). This creates a *mixture of experts* type model where the experts each represent a local approximation of the function. These experts can then be combined using a variety of tools. The one presented in Ref. [44] is the Robust Bayesian Committee Machine (RBCM), developed in Ref. [45] and shown to be effective when used with Gaussian Process models in Ref. [46].

When modelling the wind turbine power curve this approach also can be beneficial. It will reduce, further, the computational complexity of the model which in turn reduces computation time, also helps account for the first of two types of heteroscedasticity present in the data. That is, the power curve exhibits three distinct regimes that it smoothly transitions between; the first is the behaviour before cut-in, the second as the power output rises with increasing wind speed, and the third when the turbine is limited to its maximum output. This allows the data to be separated into a three component mixture. Unlike the work of Liu et al. [44], the data relating to each of these components can be and is defined based on *physical* prior knowledge. Additionally, around the transitions between two components data are included in each of the components to ensure smooth transitions between the GPs.

When making predictions using this model, the predictions

Please cite this article as: T.J. Rogers et al., Probabilistic modelling of wind turbine power curves with application of heteroscedastic Gaussian Process regression, Renewable Energy, https://doi.org/10.1016/j.renene.2019.09.145

from each expert must be combined — here by means of the RBCM. The means and variances which are predicted by each of the experts are aggregated as a weighted sum over the experts. The experts are combined separately for the Gaussian Processes over $f(\mathbf{x})$ and $g(\mathbf{x})$. For a committee model with C experts, each expert has a calculated predictive mean and variance for $f(\mathbf{x})$ and $g(\mathbf{x})$ which can be indexed according to which expert made that prediction. For example $\sigma^{2(g)}_{\star i}$ indicates the predictive variance of the i^{th} expert for the GP over $g(\mathbf{x})$, this has an analogous precision $\sigma^{-2(g)}_{\star i} = 1/\sigma^{2(g)}_{\star i}$.

The aggregated predictive distribution for $f_\star$ has a mean given by,

$$\mu^{(f)}_{\star,\mathscr{A}} = \sigma^{2(f)}_{\star,\mathscr{A}} \sum_{i=1}^{C} \beta_i^{(f)} \sigma^{-2(f)}_{\star i} \mu^{(f)}_{\star i} \tag{30}$$

and precision,

$$\sigma^{-2(f)}_{\star,\mathscr{A}} = \sum_{i=1}^{C} \beta_i^{(f)} \sigma^{-2(f)}_{\star i} + \left(1 - \sum_{i=1}^{C} \beta_i^{(f)}\right) \sigma^{-2(f)}_{\star\star} \tag{31}$$

where $\sigma^{-2(f)}_{\star\star}$ is the prior precision of the GP over $f(\mathbf{x})$. Similarly for the GP over the log noise variance, the aggregated mean is given by,

$$\mu^{(g)}_{\star,\mathscr{A}} = \sigma^{2(g)}_{\star,\mathscr{A}} \left[\sum_{i=1}^{C} \beta_i^{(g)} \sigma^{-2(g)}_{\star i} \mu^{(g)}_{\star i} + \left(1 - \sum_{i=1}^{C} \beta_i^{(g)}\right) \sigma^{-2(g)}_{\star\star} \mu_0 \right] \tag{32}$$

and the precision by,

$$\sigma^{-2(g)}_{\star,\mathscr{A}} = \sum_{i=1}^{C} \beta_i^{(g)} \sigma^{-2(g)}_{\star i} + \left(1 - \sum_{i=1}^{C} \beta_i^{(g)}\right) \sigma^{-2(g)}_{\star\star} \tag{33}$$

with $\sigma^{-2(g)}_{\star\star}$ is the prior precision of the GP over $g(\mathbf{x})$. It remains to decide on the weighting functions between the experts for both $f(\mathbf{x})$ and $g(\mathbf{x})$. These are the weightings $\beta_i^{(f)}$ and $\beta_i^{(g)}$ respectively. Since the GP model automatically returns a measure of uncertainty in the prediction it makes (the VSHGP included), this can be used as some measure of confidence in the prediction being made at any point. It is therefore possible to use the variance of the prediction to weight the experts. The variance for each GP is bounded by its prior variance, therefore it is possible to establish the weighting of each expert by comparing its predictive variance to its prior variance. Given this the weighting function for $f(\mathbf{x})$ is given as,

$$\beta_i^{(f)} = \frac{1}{2}\left(\log \sigma^{(f)}_{\star\star} - \log \sigma^{(f)}_{\star i}\right) \tag{34}$$

and likewise for $g(\mathbf{x})$,

$$\beta_i^{(g)} = \frac{1}{2}\left(\log \sigma^{(g)}_{\star\star} - \log \sigma^{(g)}_{\star i}\right) \tag{35}$$

Finally, once the means and variances for the predictions of each expert have been aggregated for both $f(\mathbf{x})$ and $g(\mathbf{x})$, the first two moments of the variational predictive distribution can be written down as,

$$\mathbb{E}_{q(y_\star)}[y_\star] = \mu_{\star,\mathscr{A}} = \mu^{(f)}_{\star,\mathscr{A}} \tag{36a}$$

$$\mathbb{V}_{q(y_\star)}[y_\star] = \sigma_{\star,\mathscr{A}} = \sigma^{2(f)}_{\star,\mathscr{A}} + \exp\left\{\mu^{(g)}_{\star,\mathscr{A}} + \frac{1}{2}\sigma^{2(g)}_{\star,\mathscr{A}}\right\} \tag{36b}$$

Despite the somewhat circuitous route, the similarity between these equations and those shown in equation (21) make clear that this model is merely an extension of the non-sparse heteroscedastic model. In addition to this, these equations approximate the probability distribution over the mean and variance of each output given the previously observed data, $\mathscr{D} = \{X, \mathbf{y}\}$, in a manner analogous to the standard GP. As such, without needing to pre-specify a functional form for the data, the input-output relationship — with a heteroscedastic noise model — can be learnt in a Bayesian manner, returning a probabilistic output.

3. Modelling wind turbine power curves

With a mathematical framework for learning nonlinear functions with heteroscedastic noise models in place, attention can be directed towards prediction of wind turbine power curves. This section will explore the use of the techniques presented previously for modelling. To demonstrate the usage of these techniques a sample dataset taken from an operational wind turbine is used. For confidentiality reasons the measured values of wind speed and power have been obscured by normalisation of the data. Additionally, the values stated for the cut-in and nominal speeds of the turbine are selected to be representative in the normalised space and bear no relation to the stated values on the data sheet for the turbine being considered. However, the data collected are 10-min averages from a functional SCADA system over a period of 125 weeks, and as such, this dataset represents a realistic set of measurement data.

Following their normalisation, the data are separated into three distinct sets, one for training, one for validation, and one for testing of any models which are learnt. In the results shown here comparisons are made between predictions made on the training data — that data used to learn the model — and the test data — data that remains *unseen* by the model until predictions are made, the validation data is unused in this case. Both the training and testing datasets consist of 16359 pairs of data points where the input is the measured 10-min average wind speed and the target is the measured 10-min average power. One key modelling assumption is that the function is *stationary*, i.e. the relationship between the wind speed and power output does not change over time. Considering the training and test data (collected several months apart) which are shown overlaid in Fig. 2, this assumption appears to hold across this dataset.

Although, as has been shown, the addition of a mean function to the standard GP formulation is trivial, for the heteroscedastic formulations, only the zero-mean versions have been shown. Observing the characteristic shape of the wind turbine power curve, it is clear that a constant zero mean assumption is not valid. In view of this, it is prudent to learn a parametric mean function which can be removed from the data before learning the GP model of choice. Two potential mean functions are considered, the first a piecewise-linear function that could be specified from the known *cut-in* and *nominal* speeds of the turbine, the second a hyperbolic tangent function the parameters of which must be learnt from the data. The piecewise-linear function is defined as a three-component curve, with a constant power output of zero, before the cut-in speed, and a constant of the rated power output above the nominal speed. A line then connects these two values between the cut-in and nominal speed.

Fig. 3 shows a comparison between the piecewise-linear fit and the hyperbolic tangent fit. The parameters of the hyperbolic tangent have been learnt by a minimisation of the sum of squares errors on the training data — i.e. a standard least squares fit — using a quantum particle swarm population based optimiser [47]. The requirement for a very robust fit is relaxed since the GP is expressive enough to compensate for any bias introduced by learning 'sub-optimal' parameters of this model.

The models learnt by these parametric fits have been removed

Please cite this article as: T.J. Rogers et al., Probabilistic modelling of wind turbine power curves with application of heteroscedastic Gaussian Process regression, Renewable Energy, https://doi.org/10.1016/j.renene.2019.09.145

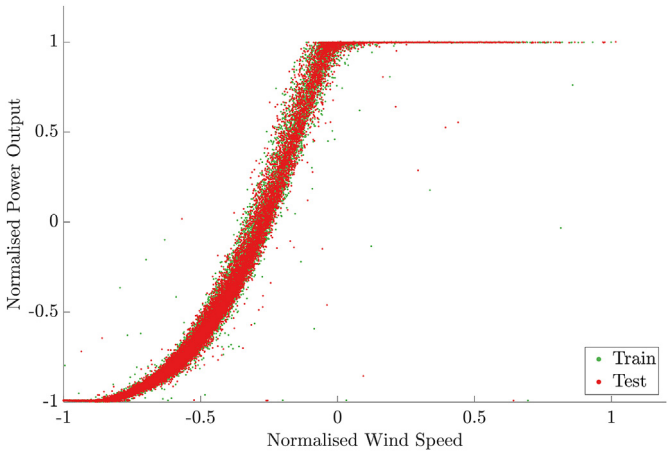

Fig. 2. Normalised training and test wind turbine power curve data.

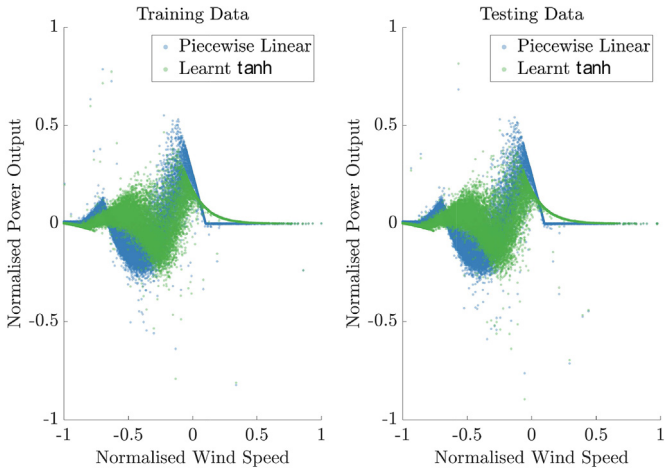

Fig. 4. The "zero" mean transform space after applying each mean (parametric) function.

from the data in order to transform it into a zero-mean space that can be used with the GP model. In Fig. 4, the data in this transformed space is shown. The piecewise linear model is seen to create a hard corner as it transitions between sections, this leaves the function to be learnt by the GP as non-smooth and discontinuous. Remembering that the choice of kernel encodes smoothness beliefs about the functions, to make use of the piecewise linear mean would require finding a covariance function that allows for non-smooth functions. However, using the hyperbolic tangent as a mean functions leads to a smoother function in the transformed space. The data, following removal of the hyperbolic tangent mean, would be more readily learnt using more common covariance functions encountered when using GP models, e.g. the squared exponential in equation (5).

Having removed this hyperbolic tangent mean, the task of fitting a GP model can begin. For comparison, it would be preferential to have fit a full homoscedastic GP, the VFE sparse GP, the full heteroscedastic, and the sparse heteroscedastic. However, due to the size of the data set, over 16000 training points, it is not possible to fit a non-sparse GP with any reasonable amount of resources. This will often be the case when modelling wind turbine power curves as data acquisition systems typically run for extended periods of time

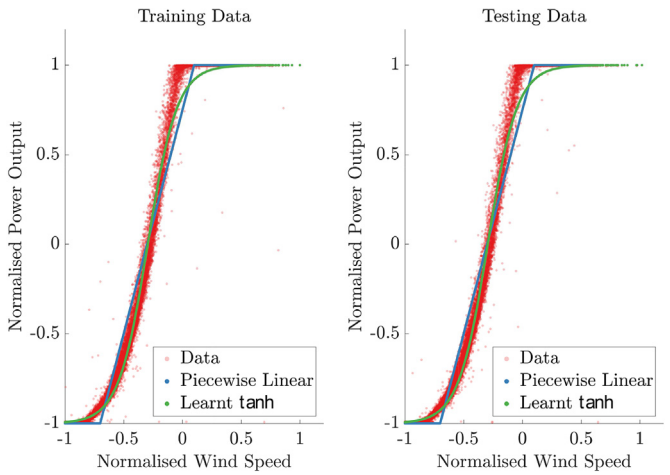

Fig. 3. Two different parametric models fitted to the power curve, the piecewise linear model (in blue) from assumed values and the hyperbolic tangent model (in green) learnt from the training data. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

accumulating very large datasets. It is also beneficial to use computationally efficient methods when inference needs to be conducted online, this may include the retraining of these models to make comparisons as the turbine ages. Therefore, given the necessity to use a sparse method, comparison is made between the VFE (a homoscedastic model) and the distributed sparse variational heteroscedastic GP (DSVHGP). Results are shown for fits to both the training and testing data with and without the hyperbolic tangent mean added.

Two quantities are used to assess the model fit, the first is a normalised mean squared error (NMSE) shown in equation (37); this measures the goodness-of-fit of point predictions of a model — either the output of a deterministic model or the mean of a probabilistic one. The NMSE will return a score of zero in the case of a perfect fit, a score of 100 corresponds to a prediction which has the same error as simply taking the mean of the data.

$$\mathrm{NMSE} = \frac{100}{N\sigma_y^2}\sqrt{(\mathbf{y}-\widehat{\mathbf{y}})^T(\mathbf{y}-\widehat{\mathbf{y}})} \tag{37}$$

Here, N is the number of datapoints being assessed, σ_y^2 the variance of the measured data, $\mathbf{y}$ the measured data, and $\widehat{\mathbf{y}}$ the predicted outputs. In addition to this, the joint likelihood of the probabilistic models is used to assess model quality. The NMSE metric fails to capture any quantification of uncertainty in the model. Since one of the main benefits of the GP approach is this automatic quantification of uncertainty, it is sensible to include this as a measure of goodness-of-fit. The joint likelihood is calculated as the product over the likelihood of each prediction, given that at each prediction a univariate Gaussian distribution is returned — in the case of the heteroscedastic model this is an approximate distribution given the first two moments of the prediction equation (21).

The predictions made by the VFE model are shown in Fig. 5 and Fig. 6, in the transformed space and on the original power curve data respectively. Before stating the quantitative assessment of this model, it can be seen that the model has failed to capture the uncertainty present in the data well. There exist many points in the rising part of the power curve which lie outside of the three sigma intervals, however, at low and high wind speeds (where the function is flat) the variance is overestimated. It would appear that, as expected, the noise variance has been learnt to be an average between the high variance section as the function rises and the low variance sections towards the edges. On first inspection it may appear that the variance has been captured well in the middle

Please cite this article as: T.J. Rogers et al., Probabilistic modelling of wind turbine power curves with application of heteroscedastic Gaussian Process regression, Renewable Energy, https://doi.org/10.1016/j.renene.2019.09.145

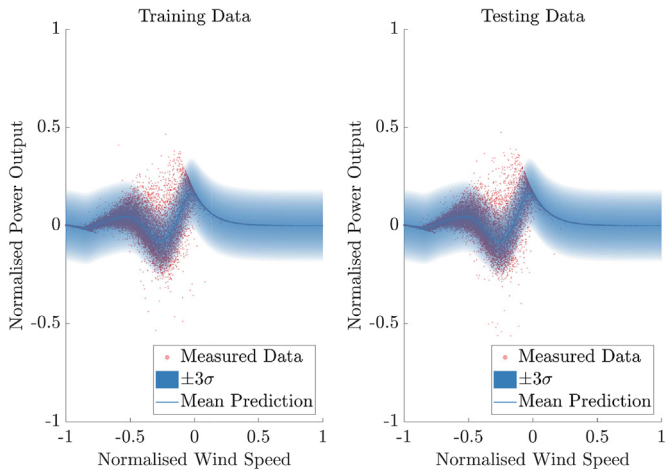

Fig. 5. Prediction made by the homoscedastic sparse GP in the transformed space.

section of the function and poorly at the end, however, on closer inspection it is seen that in addition to the overestimation of variance at high and low wind speeds the variance is actually underestimated in the middle section of the curve. With reference to the prediction in the transformed space in Fig. 5, the NMSE of the process is 56.4 for the training data and 56.2 for the test data. This is expected as the variance related to noise on the data is large in this space. Transforming back to the full power curve in Fig. 6, by adding back the hyperbolic tangent mean function, it can be seen that the fit that may look unimpressive in the transformed space actually represents the mean behaviour of the power curve well (despite poor modelling of the variance), this yields a NMSE of 0.81 for both the training and testing data. For comparison it is worth stating the NMSE scores of the two parametric functions considered. The piecewise-linear function scores 3.93 and 3.94 on the training and testing data, whereas, the hyperbolic tangent scores 1.49 and 1.50. From this it can be seen that the use of the GP with the hyperbolic tangent as a mean function leads to a significant decrease in the point-wise error. Considering the likelihoods of the models (stated as log likelihoods), in the transformed space these score 2.26×10^4 for both training and testing data. With the mean function added these scores remain the same, for both training and testing data, this allows more consistent comparison of the models as it both incorporates the uncertainty of the prediction and is insensitive to removing the parametric mean function.

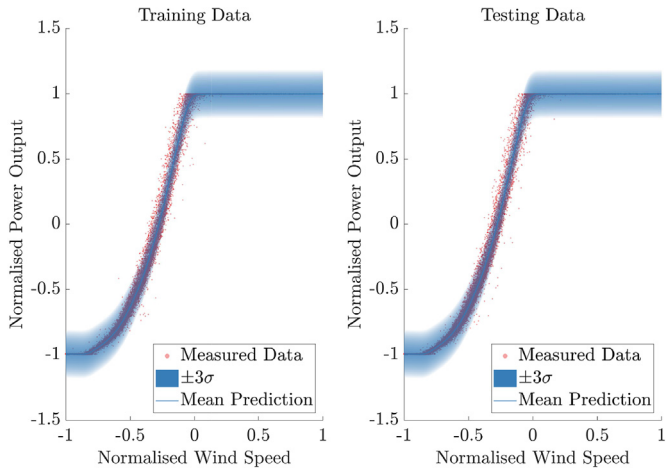

Fig. 6. Predictions of the homoscedastic sparse GP of the full power curve.

A heteroscedastic model was also learnt and tested on the same datasets. The distributed sparse variational heteroscedastic GP is chosen for this task, to allow heteroscedastic inference over this large dataset. The training data are separated in to three overlapping datasets, which are in turn, used to train three experts in the robust Bayesian committee model framework. Fig. 7 shows the predictions made by these experts when predicting the test dataset. In the upper three plots the data which have been used to train each expert is also shown. It can be seen that each expert has been trained on a subset of data which overlaps in the input space, this ensures a smooth transition between the experts in the committee model. The locations of these splits and the amount of overlap were chosen *a priori* to divide the data into the three broad regions seen in the power curve:

1. Before and through cut-in speed;
2. Transition from cut-in speed to the upper bound on nominal speed;
3. Lower bound on the nominal speed to cut-out speed.

The split locations can be chosen based upon the known cut-in and nominal speeds of the turbine and the amount of overlap is a matter of user choice, for this example the normalised splits are listed in Table 1. It was the experience of the authors that a small overlap region ensured a smooth transition between the experts, in this case the ranges of wind speed were:

As expected, each expert is most capable of making predictions close to data which has been used to train that expert and is most confident of the predictions in those regions. This confidence in the predictions is the measure used to weight the contribution of that expert as calculated in equations (34) and (35). The aggregated predictions of the model (equation (36)) for the test data are shown below the contribution of each expert in Fig. 7 with the measured test data superimposed.

The aggregated predictions made by the DSVHGP in the transformed space are shown in Fig. 8 for the training and testing data. The NMSE scores for these models are 56.5 and 56.2 for the training and test data respectively. The scores in the NMSE match very closely with the homoscedastic model fitted. Moving to the full space, shown in Fig. 9, the NMSE scores are found to be 0.81 for both the training and testing data. Scores which are identical to the homoscedastic model — up to this level of accuracy. This is, perhaps, expected considering that the predictive mean of the DSVHGP model is given by the mean of the GP over the function $f(\mathbf{x})$ which is a very similar formulation to the predictive mean equation of the homoscedastic sparse model. The main difference between the models (in a predictive sense) enters through the calculation of the variance of the predictive density.

Since the NMSE score does not depend on the predictive variance of the model, it is unsurprising that this score is largely unaffected by the changes in the model. The likelihood score of the model, however, reveals the improved quantification of the uncertainty in the prediction. As before, the joint log likelihoods in the transformed space and over the full power curve are identical up to the accuracy stated. These are 3.47×10^4 for the training data and 3.39×10^4 for the testing data. This represents a marked improvement over the scores calculated for the homoscedastic model. The increase in likelihood of the predictions would indicate that the heteroscedastic formulation has been able to better capture the variance in the data and is capable of making predictions which better represent that variance.

To visualise the difference in the fits of the two models, they are shown overlaid in Fig. 10; here, the similarity in predictive mean can be easily seen and the difference in the predictive variance is also apparent. The improved modelling of the noise on the data as a

Please cite this article as: T.J. Rogers et al., Probabilistic modelling of wind turbine power curves with application of heteroscedastic Gaussian Process regression, Renewable Energy, https://doi.org/10.1016/j.renene.2019.09.145

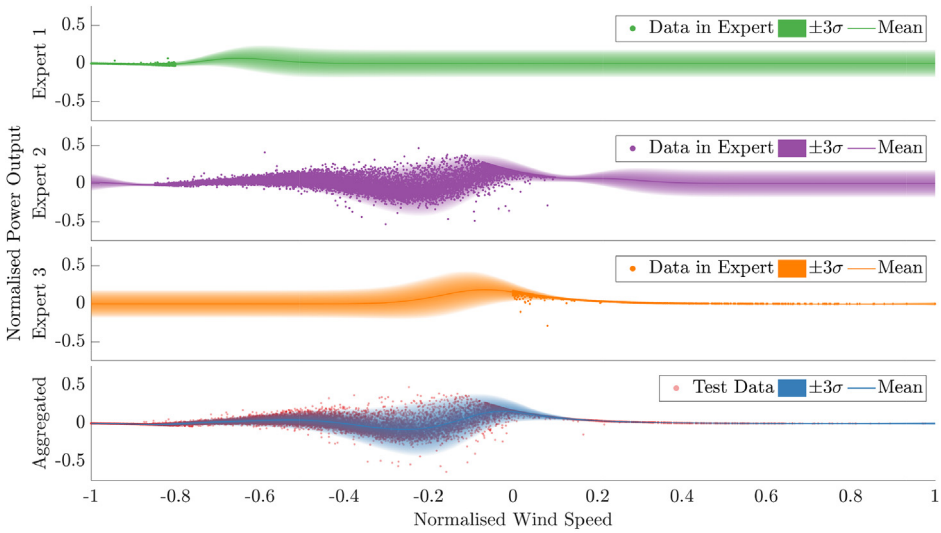

Fig. 7. Prediction of each of the experts in the robust Bayesian committee model and the aggregated prediction in the transformed space.

Table 1
Division of input space for mixture of experts.

Expert	Normalised Wind Speed	
	Lower Bound	Upper Bound
1	−1	−0.8
2	−0.85	0.1
3	0	1

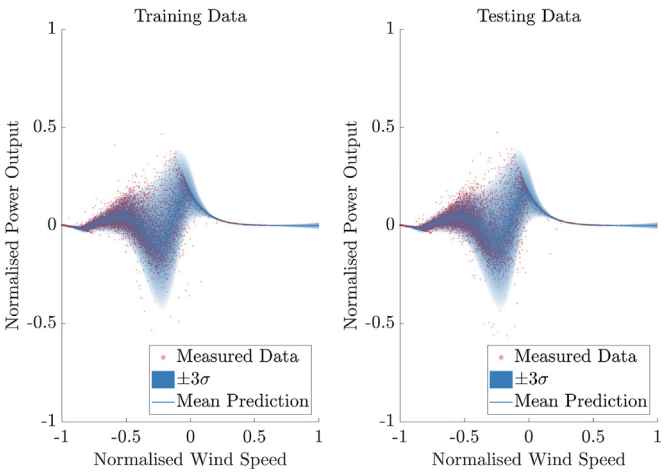

Fig. 8. Predictions of the distributed SVHGP in the transformed space.

result of the heteroscedastic GP is most apparent at the tails of the power curve where the homoscedastic model overestimates the variance. Although in certain situations this may not be of major concern, this overestimation of variance will lead to reduced sensitivity if the model is used in a damage detection setting such as in Papatheou et al. [23].

One concern that could be raised with both models, or indeed any GP fit of the power curve, would be that there is likelihood that the turbine would exceed its stated maximum power output — this would not be observed due to the limits of the turbine. This is most apparent when considering the output of the homoscedastic model since the variance of the heteroscedastic model reduces as the wind

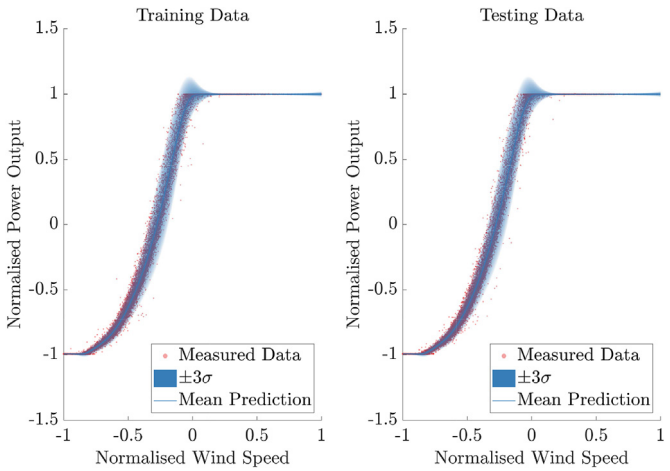

Fig. 9. Predictions of the distributed SVHGP of the full power curve.

speed increases and the turbine consistently produces its maximum rated power. However, around the nominal speed of the turbine there is variance in the power output which is captured in both models. This region is focussed on in Fig. 11, where it can be seen that the heteroscedastic model captures well the variance in power output around the nominal speed below the maximum output but has variance extending above the maximum rated output. This is an artefact of the approximation of the posterior distribution as a Gaussian based on its first two moments, although it is likely that the full distribution would also have probability mass above this maximum output. Because of the Gaussian nature of this approximate posterior, the distribution over the outputs must be symmetric about the mean. Around the nominal speed of the turbine the distribution over the power output is heavily skewed, because only the mean and variance of the distribution of the output are modelled it is not possible to model is asymmetric distribution. One solution to this is to apply prior physical knowledge to the system and to recognise that it is extremely unlikely that the turbine would exceed its rated output, therefore, in any further analysis the predictions could be limited at the maximum value. Alternative approaches to handling this issue will be discussed as future work.

Please cite this article as: T.J. Rogers et al., Probabilistic modelling of wind turbine power curves with application of heteroscedastic Gaussian Process regression, Renewable Energy, https://doi.org/10.1016/j.renene.2019.09.145

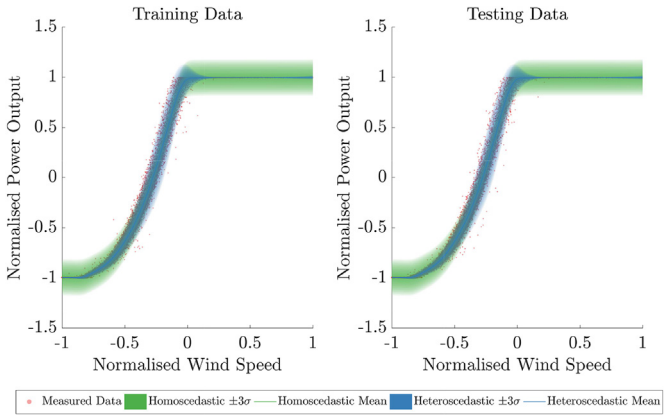

Fig. 10. Comparison of power curve predictions made by both the homoscedastic and heteroscedastic models.

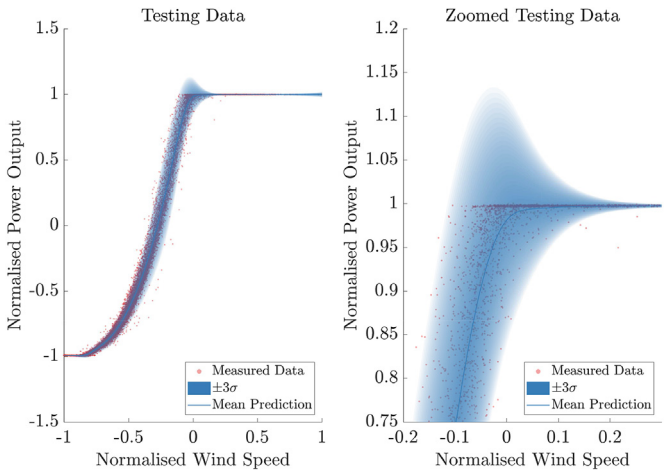

Fig. 11. Highlighting the predictions around the nominal-speed where output is limited to the rated output.

Finally, it is natural to consider the importance of each part of the modelling process. The use of the mixture of experts model to separate each region of the prediction allows for some heteroscedasticity by assigning a different noise variance to each expert. By separating the components of the prediction from the DSVHGP it is possible to inspect the role of the heteroscedastic model. In Fig. 12 the GP which accounts for the mean prediction of the model and the GP which models the log noise variance are visualised separately as well as showing the full prediction. In the top left frame of the figure it can be seen that the GP over the latent function of the mean for the power curve has a very low variance. This indicates a confident prediction of the mean behaviour of the curve, which is seen to fit well with the data. Considering the top right frame in the figure the prediction of the GP over the log noise variance is shown. Here, the role of the heteroscedastic formulation in the GP is clearly seen. If each member of the mixture of experts were to exhibit homoscedastic behaviour within its region this plot would show three horizontal lines, with quick transitions between them. Instead it can be seen that the log noise variance is itself a nonlinear function which is evolving with wind speed. This curve also shows the expected behaviour that the variance of the noise on the power curve is lower towards the upper and lower bounds on the wind speed. It can also be seen that the increase in variance close to the nominal speed of the turbine has been modelled by this

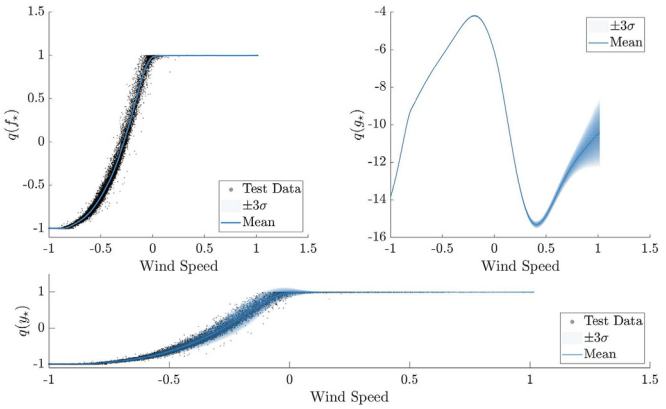

Fig. 12. Decomposition of the prediction from the mixture of experts into the GP over the mean of the process (top left) and the GP over the log noise variance of the process (top right), the combination of these two gives the full prediction shown at the bottom.

GP. Finally, due to the relatively low number of datapoints seen close to a nominal wind speed of one, the prediction of the variance in the region becomes less confident (seen by an increase in variance) and the predicted variance increases to accommodate this. However, since this is modelling the log noise variance the actual uncertainty seen in the model remains very low even toward this uncertain region since the mean remains low. Through the modelling of this collected data it has been possible to demonstrate how the proposed methodology based on the DSVHGP can accurately capture the behaviour of a power cure both in terms of its mean behaviour and through modelling the uncertainty. The use of the mixture of experts is shown to be a pragmatic approach to capturing the form of the power curve across three key regimes. Finally, the use of the heteroscedastic GP model is shown to allow the change in noise variance, across the wind speed, to be modelled; giving more reliable predictions of the uncertainty associated with the prediction across the full power curve.

4. Conclusions

The work contained in this paper has laid down a methodology for rigorous probabilistic modelling of wind turbine power curves — extending the work of [1,22,23] to the heteroscedastic case with sparsity, improving the quantification of uncertainty significantly. The Gaussian Process has been introduced as a flexible Bayesian machine learning technique for modelling of nonlinear functions. The difficultly in use of a GP model with large datasets has been discussed with the use of a sparse approximation suggested. The variational free energy approach of Titsias [42] has been presented as a powerful and robust method in which large numbers of training data points can be incorporated into the GP framework. In view of the heteroscedastic noise behaviour seen in power curve data, the extension of a GP model to include the modelling of this input dependent noise has been discussed. The method of Lázaro-Gredilla and Titsias [43] has been to shown to achieve this by modelling the log noise variance of a process as an additional Gaussian Process which is learnt in a variational manner. Combining the theory developed for the VFE sparse approximation and the heteroscedastic approach led to a sparse variational heteroscedastic GP model. This model, introduced in Liu et al. [44] allows the learning of a heteroscedastic GP over large datasets, such as those collected by SCADA systems installed on wind turbines. Finally, the approach of building a mixture of experts model based on partitioning the input space of the data is shown to lead to further computational gains and better robustness when modelling

Please cite this article as: T.J. Rogers et al., Probabilistic modelling of wind turbine power curves with application of heteroscedastic Gaussian Process regression, Renewable Energy, https://doi.org/10.1016/j.renene.2019.09.145

functions with multiple behavioural regimes. As discussed in Ref. [44] the robust Bayesian committee machine is a useful tool for doing this within a Bayesian framework.

A measured set of wind speed and power output data collected via a SCADA system has been used to demonstrate the usage of these approaches for modelling wind turbine power curves. The wind turbine power curve is seen to exhibit nonlinear behaviour and heteroscedastic noise processes. It is shown how the power curve can be transformed to approach a zero-mean space via a prelearnt mean function. The hyperbolic tangent function is used in this work since it ensures smoothness of the function to be learnt by the GP in the transformed space. Fitting the data in this space via either the homoscedastic or heteroscedastic GP leads to nearly identical NMSE scores indicating that the mean fits of the models are very similar. The models of the full power curves for both the homoscedastic and heteroscedastic GPs and for both training and testing data are found to be 0.81 — heuristically this represents a "very good" fit as the NMSE can be thought of as similar to a percentage error.

The move to heteroscedastic modelling of power curves, although having little effect on the mean prediction quality of the model and leads to far better quantification of the variance in the data. This is reflected in the likelihood scores of the predictions. The joint log likelihood of the predictions for both the training and testing data increase by over 50% when moving the heteroscedastic model. It is also seen that visually, the variance in the data is captured far better (Fig. 10).

In this work, it has been shown that the wind turbine power curve is well suited to being modelled via a heteroscedastic GP regression and the distributed sparse variational heteroscedastic GP is a powerful and expressive model with which to do this. The use of this model naturally handles the heteroscedastic noise present in the data, automatically returns predictions as the (approximate) distribution over possible outputs, and avoids the risk of overfitting — present in high-order polynomial models. Therefore, the use of this model represents a good choice should a user wish to accurately model the power curve of a wind turbine (with quantification of the uncertainty) and becomes more valuable as the probabilistic outputs are carried into further calculations.

As previously discussed, in possession of this probabilistic model, it is now possible to refine further analyses. This includes better quantification of uncertainty in SHM applications leading to reductions in false alarms and increased sensitivity to damage. It also provides important information for making macro-level decisions about the turbine or farm. This process could include the propagation of distributions over wind speed to give a distribution over expected power which can be used for better financial planning or for grid-level power management.

4.1. Future work

While the approach adopted in this paper has been seen to be effective at modelling the behaviour of a wind turbine power curve, it opens up a number of avenues of further investigation. It has been observed that, despite the mean predictions being consistent with the physical behaviour of the wind turbine, the predictive variance of the model places probability mass above the maximum power output of the turbine. This is a consequence of considering the variance of the prediction to be symmetric around the mean. While it would be possible to truncate this distribution above the maximum rated output, future work into modifications of the likelihood function could lead to more statistically robust solutions. It is also beneficial to consider if the parameters of the mean function would be better learnt inside the GP framework rather than applying the transformation into the zero-mean space before learning the (hyper)-parameters of the GP. Finally, it will be valuable in future to demonstrate the propagation of the predictive variance of the model into further analyses which may benefit from a Bayesian treatment. For example, to predict distributions over expected income from a particular turbine, or to enhance previously presented damage detection strategies.

In conclusion, the move towards nonparametric modelling of wind turbine power curves, allows the use of probabilistic models which offer robust and accurate mean predictions, as well as automatic quantification of uncertainty. The use of these models opens up better understanding of the uncertainty of the power output, of the turbine and avoids issues in overfitting that may occur in parametric models. For this reason, the use of heteroscedastic Gaussian Process models is a powerful and sensible approach moving forward.

Acknowledgements

The authors wish to acknowledge Vattenfall Wind Energy for kindly providing the data used in the results of this paper. Additionally, the work here was supported by EPSRC grant numbers EP/S001565/1, EP/J016942/1, EP/R006768/1, and EP/R004900/1.

References

[1] E. Papatheou, N. Dervilis, A.E. Maguire, I. Antoniadou, K. Worden, A performance monitoring approach for the novel Lillgrund offshore wind farm, IEEE Trans. Ind. Electron. 62 (10) (2015) 6636–6644.
[2] K. Worden, C.R. Farrar, G. Manson, G. Park, The fundamental axioms of structural health monitoring, Proc. R. Soc. A Math. Phys. Eng. Sci. 463 (2082) (2007) 1639–1664.
[3] C.R. Farrar, K. Worden, Structural Health Monitoring: A Machine Learning Perspective, John Wiley & Sons, 2012.
[4] V. Thapar, G. Agnihotri, V.K. Sethi, Critical analysis of methods for mathematical modelling of wind turbines, Renew. Energy 36 (11) (2011) 3166–3177.
[5] C. Carrillo, A.O. Montaño, J. Cidrás, E. Díaz-Dorado, Review of power curve modelling for wind turbines, Renew. Sustain. Energy Rev. 21 (2013) 572–581.
[6] M. Lydia, S.S. Kumar, A.I. Selvakumar, G.E.P. Kumar, A comprehensive review on wind turbine power curve modeling techniques, Renew. Sustain. Energy Rev. 30 (2014) 452–460.
[7] M. Marčiukaitis, I. Žutautaitė, L. Martišauskas, B. Jokšas, G. Gecevičius, A. Sfetsos, Non-linear regression model for wind turbine power curve, Renew. Energy 113 (2017) 732–741.
[8] M.M. Raj, M. Alexander, M. Lydia, Modeling of wind turbine power curve, in: ISGT2011-India, IEEE, 2011, pp. 144–148.
[9] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
[10] D. Barber, Bayesian Reasoning and Machine Learning, Cambridge University Press, 2012.
[11] E. Taslimi-Renani, M. Modiri-Delshad, M.F.M. Elias, N.A. Rahim, Development of an enhanced parametric model for wind turbine power curve, Appl. Energy (177) (2016) 544–552.
[12] S. Seo, S.-D. Oh, H.-Y. Kwak, Wind Turbine Power Curve Modeling Using Maximum Likelihood Estimation Method, Renewable Energy, 2018.
[13] D. Villanueva, A.E. Feijóo, Reformulation of parameters of the logistic function applied to power curves of wind turbines, Electr. Power Syst. Res. 137 (2016) 51–58.
[14] M. Lydia, S.S. Kumar, A.I. Selvakumar, G.E.P. Kumar, Wind farm power prediction based on wind speed and power curve models, in: Intelligent and Efficient Electrical Systems, Springer, 2018, pp. 15–24.
[15] S. Li, D.C. Wunsch, E.A. O'Hair, M.G. Giesselmann, Using neural networks to estimate wind turbine power generation, IEEE Trans. Energy Convers. 16 (3) (2001) 276–282.
[16] A. Marvuglia, A. Messineo, Monitoring of wind farms' power curves using machine learning techniques, Appl. Energy 98 (2012) 574–583.
[17] F. Pelletier, C. Masson, A. Tahan, Wind turbine power curve modelling using artificial neural network, Renew. Energy 89 (2016) 207–214.
[18] G. Ciulla, A. D'Amico, V. Di Dio, V.L. Brano, Modelling and analysis of real-world wind turbine power curves: assessing deviations from nominal curve by neural networks, Renew. Energy 140 (2019) 477–492.
[19] T. Ouyang, A. Kusiak, Y. He, Modeling wind-turbine power curve: a data partitioning and mining approach, Renew. Energy 102 (2017) 1–8.
[20] J. Yan, H. Zhang, Y. Liu, S. Han, L. Li, Uncertainty estimation for wind energy conversion by probabilistic wind turbine power curve modelling, Appl. Energy 239 (2019) 1356–1370.
[21] Y. Wang, Q. Hu, D. Srinivasan, Z. Wang, Wind power curve modeling and wind power forecasting with inconsistent data, IEEE. Trans. Sustain. Energy 10 (1)

(2018) 16–25.
[22] I. Antoniadou, N. Dervilis, E. Papatheou, A.E. Maguire, K. Worden, Aspects of structural health monitoring and condition monitoring of offshore wind turbines, Phil. Trans. R. Soc. A: Math. Phys. Eng. 373 (2035) (2015) 20140075.
[23] E. Papatheou, N. Dervilis, A.E. Maguire, C. Campos, I. Antoniadou, K. Worden, Performance monitoring of a wind turbine using extreme function theory, Renew. Energy 113 (2017) 1490–1502.
[24] B. Manobel, F. Sehnke, J.A. Lazzús, I. Salfate, M. Felder, S. Montecinos, Wind turbine power curve modeling based on Gaussian processes and artificial neural networks, Renew. Energy 125 (2018) 1015–1020.
[25] C.A. Micchelli, Y. Xu, H. Zhang, Universal kernels, J. Mach. Learn. Res. 7 (Dec) (2006) 2651–2667.
[26] R.M. Neal, Priors for infinite networks, in: Bayesian Learning for Neural Networks, Springer, 1996, pp. 29–53.
[27] J. Lee, Y. Bahri, R. Novak, S. Schoenholz, J. Pennington, J. Sohl-dickstein, Deep neural networks as Gaussian processes, in: Proceedings of the Sixth International Conference on Learning Representations, 2018.
[28] R.K. Pandit, D. Infield, J. Carroll, Incorporating air density into a Gaussian process wind turbine power curve model for improving fitting accuracy, Wind Energy 22 (2) (2019) 302–315.
[29] R.K. Pandit, D. Infield, A. Kolios, Comparison of advanced non-parametric models for wind turbine power curves, IET Renew. Power Gener. 13 (9) (2019) 1503–1510.
[30] A. O'Hagan, J. Kingman, Curve fitting and optimal design for prediction, J. R. Stat. Soc. Ser. B (1978) 1–42.
[31] C.E. Rasmussen, C.K.I. Williams, Gaussian Processes for Machine Learning, The MIT Press, 2005.
[32] A. Gelman, J.B. Carlin, H.S. Stern, D.B. Dunson, Bayesian Data Analysis 3 (2014).
[33] D.J. MacKay, Bayesian Methods for Adaptive Models, PhD thesis, California Institute of Technology, 1992.
[34] C.E. Rasmussen, Z. Ghahramani, Occam's razor, in: Advances in Neural Information Processing Systems, 2001, pp. 294–300.
[35] I. Murray, Z. Ghahramani, A Note on the Evidence and Bayesian Occam's Razor, 2005.
[36] A. Solin, S. Särkkä, Hilbert Space Methods for Reduced-Rank Gaussian Process Regression, 2014. arXiv:1401.5508.
[37] J. Hensman, N. Durrande, A. Solin, et al., Variational Fourier features for Gaussian processes, J. Mach. Learn. Res. 18 (151–1) (2017).
[38] J. Quiñonero-Candela, C.E. Rasmussen, A unifying view of sparse approximate Gaussian process regression, J. Mach. Learn. Res. 6 (Dec) (2005), 1939–1959.
[39] T.D. Bui, J. Yan, R.E. Turner, A unifying framework for sparse Gaussian process approximation using power expectation propagation, Stat 23 (1050) (2016).
[40] M. Bauer, M. van der Wilk, C.E. Rasmussen, Understanding probabilistic sparse Gaussian process approximations, Adv. Neural Inf. Process. Syst. (2016) 1533–1541.
[41] A.G.d.G. Matthews, Scalable Gaussian Process Inference Using Variational Methods, PhD thesis, University of Cambridge, 2017.
[42] M. Titsias, Variational learning of inducing variables in sparse Gaussian processes, Artif. Intell. Stat (2009) 567–574.
[43] M. Lázaro-Gredilla, M.K. Titsias, Variational heteroscedastic Gaussian process regression, in: ICML, 2011, pp. 841–848.
[44] H. Liu, Y.-S. Ong, J. Cai, Large-scale heteroscedastic regression via Gaussian process. arXiv:1811.01179, 2018.
[45] V. Tresp, A Bayesian committee machine, Neural Comput. 12 (11) (2000) 2719–2741.
[46] M. Deisenroth, J.W. Ng, Distributed Gaussian processes, in: International Conference on Machine Learning, 2015, pp. 1481–1490.
[47] J. Sun, B. Feng, W. Xu, Particle swarm optimization with particles having quantum behavior, in: Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No. 04TH8753), vol. 1, IEEE, 2004, pp. 325–331.

Please cite this article as: T.J. Rogers et al., Probabilistic modelling of wind turbine power curves with application of heteroscedastic Gaussian Process regression, Renewable Energy, https://doi.org/10.1016/j.renene.2019.09.145