Mechanical Systems and Signal Processing 169 (2022) 108530

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

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

Bayesian modelling of multivalued power curves from an operational wind farm

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

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

ARTICLE INFO

Communicated by S. Fassois

Keywords:
Wind energy
Power curve
Performance monitoring
Structural health monitoring
Gaussian processes
Power curtailment

ABSTRACT

Power curves capture the relationship between wind speed and output power for a specific wind turbine. Accurate regression models of this function prove useful in monitoring, maintenance, design, and planning. In practice, however, the measurements do not always correspond to the ideal curve: power curtailments will appear as (additional) functional components. Such multivalued relationships cannot be modelled by conventional regression, and the associated data are usually removed during pre-processing. The current work suggests an alternative method to infer multivalued relationships in curtailed power data. Using a population-based approach, an overlapping mixture of probabilistic regression models is applied to signals recorded from turbines within an operational wind farm. The model is shown to provide an accurate representation of practical power data across the population.

1. Introduction

Given an increased demand for renewable energy, accurate predictive models are essential to justify, manage, and monitor wind turbine power generation. In particular, accurate predictions of the *power output* (under uncertainty) enable reliable forecasting of the expected income for a complete wind farm – as well as individual turbines – to support the expansion of wind-based energy [1]. Robust models of the power output have potential applications in performance monitoring and operator control, to ensure optimal use *in situ* [2,3].

Power curves capture the relationship between wind speed and turbine output power [2] – the associated function can be used as a key indicator of performance [4]. A regression can be inferred to approximate the relationship given operational measurements (training data) – typically recorded using Supervisor Control and Sensory Data Acquisition (SCADA) systems [3]. An example of SCADA data is presented in Fig. 1; a regression of these data should generalise to future measurements given *optimal* operation of the wind turbine.

Various techniques have been proposed to model training data that correspond to *ideal* operation [5–7]. In practice, however, only a subset of measurements will typically represent this relationship. In particular, power *curtailments* will appear as additional functional components; these usually correspond to the output power being controlled (or limited) by the operator. Reasons to limit power include: requirements of the electrical grid [8,9], the mitigation of loading/wake effects [10], and restrictions enforced by planning regulations.

* Corresponding author.
E-mail address: lbull@turing.ac.uk (L.A. Bull).

https://doi.org/10.1016/j.ymssp.2021.108530
Received 13 November 2020; Received in revised form 4 August 2021; Accepted 10 October 2021
Available online 25 November 2021

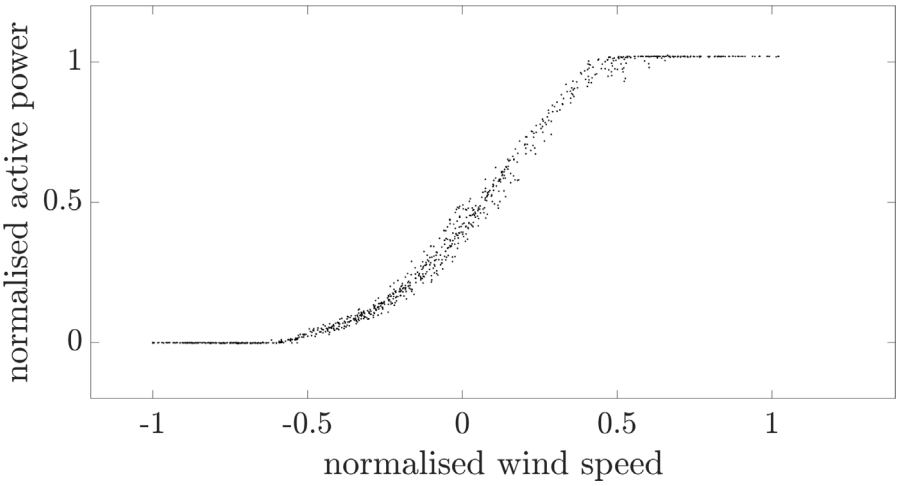

Fig. 1. Data that represent an ideal power curve. Measurements from three turbines over a period of three weeks.

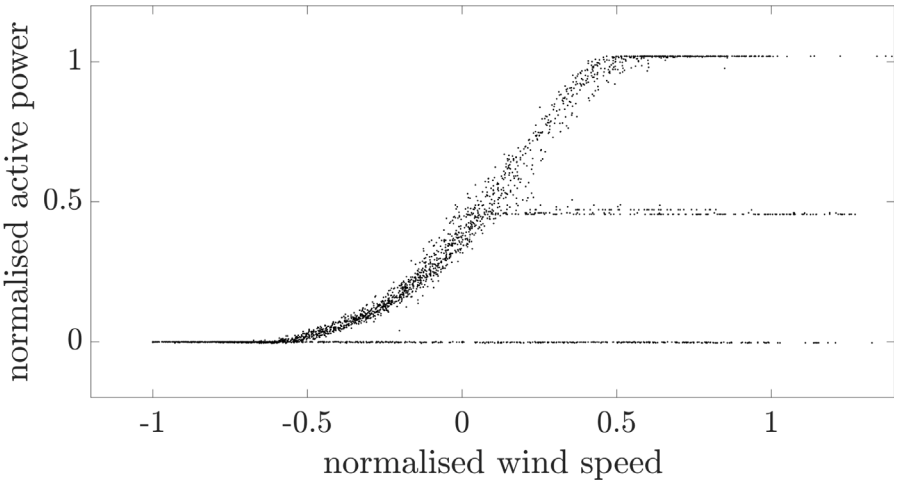

Fig. 2. Data including power curtailments — corresponding to (i) the ideal power curve (ii) ≈50%-limited output, and (iii) zero-limited output. Measurements from seven turbines over nine weeks.

An example of operational data including curtailments is shown in Fig. 2. The emergent space is multivalued, differing significantly from the archetypal curve in Fig. 1 (additionally, it cannot be modelled by conventional regression). Typically, the curtailment data are removed during pre-processing via engineering judgement [2], alongside filtering [11,12] and outlier analysis [13,14] (see Section 2.1 for details).

Disregarding curtailment data is logical when modelling the *ideal* curve, corresponding to optimal operation [4]; despite this fact, curtailed observations are expected in practice. Therefore, a representation of *in situ* measurements should model these data, rather than filtering them out, particularly in monitoring or forecasting applications (outlined in Section 2.1).

The current work suggests an overlapping mixture of probabilistic regression models [15] (i.e. Gaussian processes [16]) to infer multivalued power curves — such as those in Fig. 2. The statistical method can represent operational power data, including curtailments, while negating requirements for user annotation of the observed data — i.e. categorisation of curtailments is unsupervised. As a result, the model can represent observations that might be recorded from *in-situ* turbines in operation (rather than the ideal case only), without the need for extensive outlier analysis, filtering, or pre-processing.

2. Related work

This work relates to existing literature (e.g. [2,3,14]) concerning performance monitoring and prediction via wind turbine power curves. As aforementioned, numerous data-based models have been investigated, many of which have been summarised in review papers [5–7]. A brief summary is provided.

Parametric methods fit parametrised functions to power curve data; some examples include polynomials and sigmoid (tanh/logistic) functions [1,7,13]. Parametric models are desirable – sigmoid-type functions in particular – as properties that appear inherent to power curves can be included; for example: the cut-in/cut-out wind speeds, bounded power above and below these values, as well as *near-linear* behaviour within the bounds. Unfortunately, over-simplified functions can prove restrictive when approximating the wind-power relationship, while overly complex models (e.g. high-order polynomials) are susceptible to overtraining, and require validation procedures to ensure good generalisation to new data [4].

Alternative methods consider the data alone, and, in general, do not incorporate prior engineering knowledge. Some examples include multilayer perceptions [11], random forests [17], and support vector machines [18]. While these tools have proved effective in various machine learning tasks, many require stringent validation procedures, as the flexibility of the algorithms can easily lead to over-parametrised models in wind turbine applications — as discussed in [4].

To combat the issues of overtraining, one option considers Gaussian Process (GP) regression [2,4,14,16]. GPs relieve the need for validation as they are naturally self-regularising through the Bayesian Occam's razor [19]; that is, training/optimisation will find the minimally-complex model given the observations in the training set. While GPs are typically referred to as nonparametric, a parametrised mean function (e.g. a sigmoid function) can be defined in the *prior* of the model. In general terms, the Bayesian formulation allows for the natural inclusion of engineering knowledge of the expected functions, without the need to specify the function directly [16]. As a result, GP regression can be viewed as a middle ground between purely data-based methods, and those that are based on engineering knowledge.

2.1. Power-curtailments

There is an established literature concerning power curtailments in wind energy; for example, [8–10,20,21]. Generally, the literature considers physics-based simulation techniques for prediction, or control procedures to *enforce* curtailment — as opposed to data-driven models of wind-power measurements. For example, Hur and Leithead [9] present a wind farm controller to adjust the power generated by turbines while considering the requirements of the grid (for a simulated wind farm). It is shown that, by considering the entire wind farm in a control system, the output-power can be curtailed more effectively, such that turbines with high wind-speeds compensate for those with lower wind-speeds. Bontekoning et al. [10] present an algorithm to determine the available power of a wind farm during curtailment, when considering the *reduced wake-effect*. This phenomenon occurs when a turbine is curtailed, leading to a reduced wake for downstream turbines; in turn, this leads to an apparent increase in the available power. A physics-based model is used to adjust calculations of the available power during curtailment interactions. A number of papers (e.g. Fan et al. [20], Luo et al. [21]) have analysed the history of power-curtailments for wind energy in China, to establish potential solutions and improve the utilisation of the available resources. A range of technical, planning, and policy-making strategies are proposed, highlighting the importance of understanding the expected curtailments when planning wind farm projects.

For data-driven power curve models, the curtailment data are typically considered as outliers, and removed during pre-processing; this is because the typical concern is to characterise *ideal* operation. While the removal of these data makes a regression model simpler, the outlier analysis is non-trivial; for example, Manobel et al. [11] flag and remove outliers using a threshold based on a Gaussian Process regression, while Marvuglia and Messineo [12] de-noise the data using kernel principal component analysis. Alternatively, Marčiukaitis et al. [13] use the quartile/interquartile range over windowed inputs to detect and remove outliers, while Papatheou et al. [14] use *labels* for weekly subsets of data, provided by an expert, to remove measurements that do not correspond to ideal operation.

2.2. Why model power-curtailments?

While it is logical to remove curtailment data when modelling an ideal wind-power relationship, it is desirable to consider these 'outliers' in critical applications — namely, *monitoring*, and *forecasting*. In data-driven monitoring [22], the model should approximate all the variations of the permitted *normal* condition to inform reliable novelty detection. If the model represents ideal operation only, measurements corresponding to acceptable curtailments (via control interactions) will be flagged as abnormal. Such a monitoring regime would lead to a large number of false positives; a recognised issue in the turbine monitoring literature [3]. On the other hand, accurate curtailment modelling should prove useful within reliable forecasting frameworks. That is, if a model considers all of the expected measurements *in-situ*, it should be more informative than a model of ideal-operation only; i.e. power predictions prove more conservative if curtailment data are considered. It should be noted, however, that the proposed model can only approximate curtailments that have been previously observed.

Finally, if curtailment data are modelled rather than removed, they can be naturally separated using the model itself, instead of a separate outlier analysis procedure. As discussed, the process of outlier removal proves far from trivial [11–14].

3. Contribution

A novel algorithm is proposed to *model* curtailments in wind turbine power curves. The method offers an alternative to the conventional approach, which filters out the associated data. The algorithm expands on previous work concerning Gaussian processes (GP) [2,4,14] by inferring an overlapping mixture-model of GP components — introduced by Lázaro-Gredilla et al. [15]. An alternative (parametrised) mean function is suggested (for the GPs) that is scalable, and therefore suited to represent the expected functions for curtailed data. This choice of mean function allows for the inclusion of prior engineering knowledge and leads to interpretable hyperparameters. For each component (i.e. power curve) in the mixture of regression models, input-dependent (heteroscedastic) noise is approximated (according to Kersting et al. [23]), an important consideration for probabilistic models of wind turbine power data [4].

3.1. Layout

Section 4 introduces the SCADA dataset and the issues associated with modelling operational (curtailed) measurements. Section 5 summarises conventional Gaussian process regression for power curve modelling and introduces a novel parametrised mean function, as well as methods to approximate input-dependent noise for curtailment data. Section 6 describes the Overlapping Mixture of Gaussian Processes (OMGP) for power curve modelling, combined with ideas from Section 5. Section 7 applies the model to *in situ* operational SCADA data, and proposes methods for population-based monitoring with the OMGP. Section 8 offers concluding remarks.

4. Operational wind farm data: Population-based monitoring

This work considers a SCADA dataset, recorded from an operational wind farm owned by Vattenfall, originally presented in [2]. For confidentiality reasons, information regarding the specific type, location, and number of turbines cannot be disclosed. The data were recorded from a farm containing the same model of turbine, over a period of 125 weeks [2,14]. Observations consist of the mean *power* produced and measured *wind speed* over ten minute intervals. Sub-samples of this dataset are shown in Figs. 1 and 2.

Primarily, the suggested method considers a population-based approach to performance monitoring — associated with population-based structural health monitoring (PBSHM) [24–26]. That is, data from a population (the wind farm) are considered to infer a model (the power curve) that is representative of the group — this general model is referred to as the *form* in PBSHM [24]. To reiterate: robust and accurate models of *in situ* population data are required to monitor the wind farm.

4.1. Dataset details

For the SCADA data analysed in this work, the observations are *unlabelled*; i.e. records of the operational, environmental, or damage condition are not available. Considering Fig. 2, this fact implies that there is no ground truth to indicate which underlying function generated each sample: (i) normal operation, (ii) ≈50% curtailment, (iii) or zero-power.[1] As such, when modelling the curtailments, labels to associate data with wind-power relationships (i–iii) are unobserved and must be represented as latent variables. It is important to note: if labels were available (in a control log, for example) they should be *observed* variables in the model. In the absence of labelling for functions (i-iii), the model must allocate observations in an *unsupervised* manner, which proves non-trivial (consider outlier analysis procedures from previous work [11–14]).

To clarify, weekly subsets of data are presented in Fig. 3; notice that each set can be associated with more than one operational condition (i-iii). While separate trends are visually clear, manually labelling each point with the *ground-truth* is infeasible. For example, it is clear that data represent normal (i) and 50% curtailment (ii) in the left of Fig. 3; however, it becomes difficult to assign measurements to functions as they overlap. Likewise, while certain data clearly correspond to zero-power (iii) in the right of Fig. 3, it is unclear if the remaining data correspond to 50% curtailment (ii) or normal operation (i).

Conveniently, the labels can be modelled as a latent random variable. In turn, a predictive distribution can associate ‘soft-labels’ with the data, such that a (non-zero) likelihood associates measurements with each of the underlying functions (i-iii).

4.2. Data selection

While this work aims to represent more realistic measurements from an operational wind farm, it should be clarified that preprocessing steps are still required. The study here primarily considers data from a subset of seven turbines over a period of nine weeks (as well as four alternative turbines over seven weeks, for validation). Very sparse outliers are removed via a standard K-nearest-neighbour approach [27]. The subsets of data were selected as they contain three trends of data (i-iii). It is acknowledged, however, that alternative curtailments can occur, relating to different levels of limited power. Some examples of alternative functions from different turbines are demonstrated in Section 7.1.

5. Gaussian processes to model curtailed power curves

Before introducing the overlapping mixture model (as well as heteroscedastic updates) it is useful to summarise conventional GP regression. In this application, wind speed measurements correspond to the inputs x_i, while power measurements correspond to the outputs y_i. Given a set of N training data, $D = \{x_i, y_i\}_{i=1}^{N} = \{\mathbf{x}, \mathbf{y}\}$, the predictive distribution of the power output y_* for a new measurement of wind speed x_* is inferred. Following a probabilistic approach, the power curve is modelled by some noiseless latent function $f(x_i)$, plus an independent noise term ϵ_i,

$$y_i = f(x_i) + \epsilon_i \tag{1}$$

Rather than inferring the parameters of a function f (as with conventional parametric regression) a GP prior is placed over the functions directly. A Gaussian prior is also assumed for the noise term ϵ_i (the other latent variable). Using a Bayesian framework,

[1] While the zero-power trend is not a typical curtailment, is it considered here as a function whose data are typically filtered out before modelling. Additionally, the data are interesting to consider, as they differ functionally from other trends in the measurements.

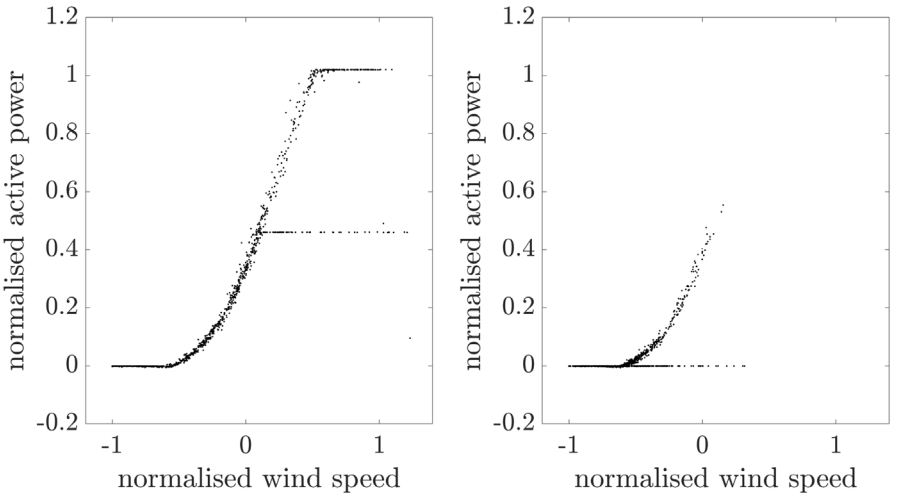

Fig. 3. Examples of weekly data subsets, measured from individual turbines.

a posterior distribution over the expected functions can be obtained, once training data D have been observed. The GP prior is defined by its mean $m(x_i)$ and covariance function $k(x_i, x_j)$; while the Gaussian prior is parametrised by σ,

$$f(x_i) \sim GP\left(m\left(x_i\right), k\left(x_i, x_j\right)\right) \tag{2}$$

$$\epsilon_i \sim \mathcal{N}(0, \sigma^2) \tag{3}$$

Over a finite and arbitrary set of inputs $\mathbf{x} = \left\{x_1, \ldots, x_N\right\}$, the GP is a (joint) multivariate Gaussian [27],

$$p(\mathbf{f} \mid \mathbf{x}) = \mathcal{N}(\mathbf{m}, \mathbf{K_{xx}}) \tag{4}$$

where $\mathbf{m} = \{m(x_i), \ldots, m(x_N)\}$ and $\mathbf{f} = \{f(x_i), \ldots, f(x_N)\}$, while $\mathbf{K_{xx}}$ is the covariance matrix, such that $\mathbf{K_{xx}}[i, j] = k(x_i, x_j) \quad \forall i, j \in \{1, \ldots, N\}$. Note: square brackets are used to index matrices and vectors when subscripts become cluttered.

Importantly, via the mean $m(x_i)$ and covariance $k(x_i, x_j)$, the GP prior can be used to encode knowledge of the expected functions given engineering judgement (before data are observed). The covariance function determines the correlation between outputs y_i and y_j – it determines properties such as the process variance, and smoothness [15]. A popular (and relatively interpretable) choice of $k(\cdot)$ is the squared-exponential function (which is used here),

$$k(x_i, x_j) = \sigma_f^2 \exp\left\{-\frac{1}{2l^2}(x_i - x_j)^2\right\} \tag{5}$$

where σ_f is the process variance, defining variance of the expected functions about the mean, and l is the length scale, which determines the rate at which the correlation between outputs decays across the input space (smoothness).

Since the GP is flexible enough to model *arbitrary* trends [27], a zero-mean function is typically assumed [15,16] such that $m(x_i) = 0$; this is usually (somewhat) justified by subtracting the sample mean and standard deviation from the outputs $\mathbf{y}$. However, if knowledge of the *expected* functions can be encoded via an explicit/parametrised mean (even approximately) this should be included [24][2]. With an explicit mean, the resulting algorithm can be considered *semi-parametric* [27], such that the GP models the residuals between the data and some parametrised function $m(x_i)$ (i.e. the prior mean).

5.1. Prior knowledge of the expected functions

As aforementioned, sigmoid functions can be used to *approximate* the expected power curve relationship: they exhibit a near-linear relationship within bounds (cut-in cut-out wind speeds) and horizontal asymptotes (re. min/max power) for high and low inputs ($x_i \to \pm\infty$). Sigmoids have been applied to power curves in the past — for parametric regression, e.g. [1,7,13], as well as within GPs [4]. A scaled version of the soft-clip (SC) function, presented by Klimek and Perelstein [28], is suggested as an alternative for this application,

$$m(x_i\ ;\ \beta, \boldsymbol{\alpha}) = \frac{\alpha_1}{\beta} \log\left\{\frac{1 + e^{\beta v}}{1 + e^{\beta(v-1)}}\right\} \tag{6}$$

$$v \triangleq \alpha_2 x_i + \alpha_3$$

[2] It is acknowledged, however, that a poor choice of prior can lead to inferior predications.

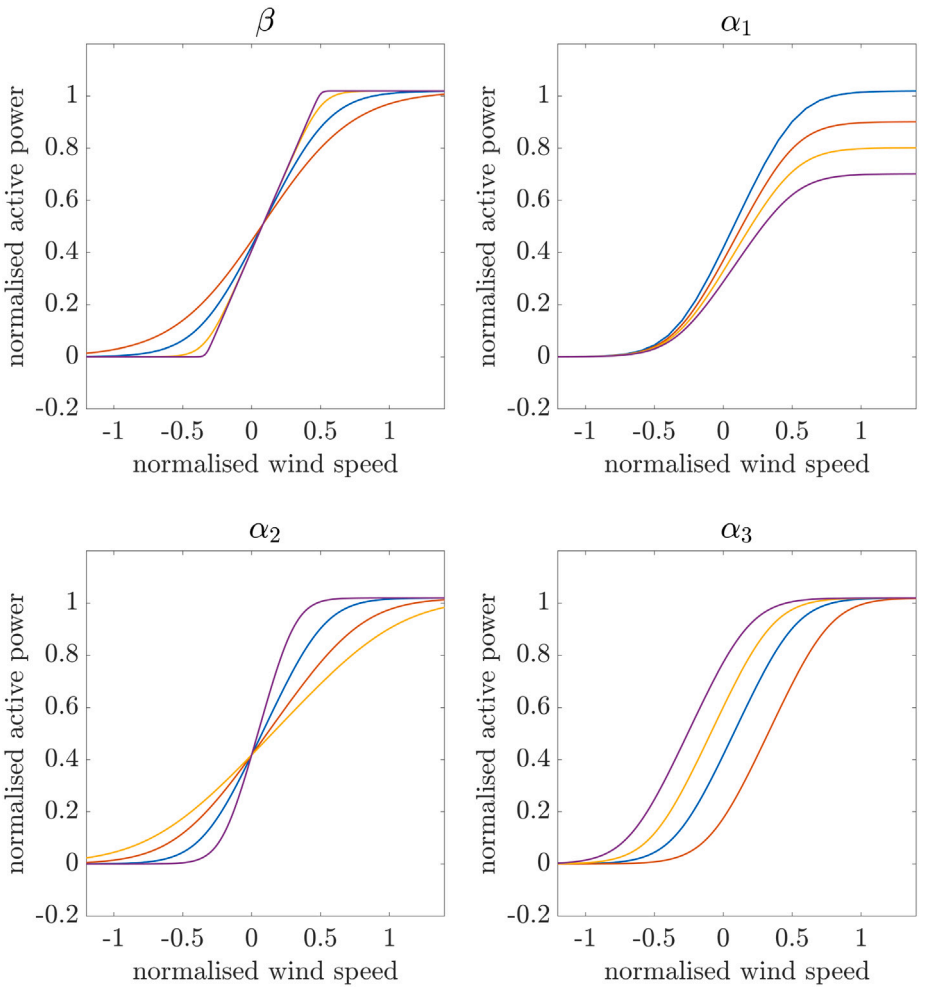

Fig. 4. Effects of the hyperparameters on the mean function of the prior $m(x_i;\beta,\boldsymbol{\alpha})$.

$$\boldsymbol{\alpha} \triangleq \{\alpha_1, \alpha_2, \alpha_2\} \tag{7}$$

Relating to power curves, the hyperparameters $\{\beta,\boldsymbol{\alpha}\}$ are interpretable. α_1 determines the value of the horizontal (non-zero) asymptote, which corresponds to the maximum (or limited) power. β controls the *rate* at which the near-linear section tends to the asymptotic values (around the cut-in/cut-out wind speed). Finally, α_2 *scales* and α_3 *translates* the function with respect to the x_i axis. Fig. 4 illustrates the effects of $\{\beta,\boldsymbol{\alpha}\}$. Importantly, control of the convergence rate via β is particularly useful for curtailed data. Consider the ≈50% limited trend in Fig. 2: a sigmoid approximation would need to be scaled, such that $\alpha_1 \approx 0.5$, while β must also increase to define sharper asymptotic behaviour. It is acknowledged that the zero-power trend (visible in Fig. 2) does not resemble a soft-clip function. In fact, a linear regression would approximate these data – a suitable component is introduced in Section 6.

5.2. Prediction and optimisation

The collected hyperparameters of the model (associated with the mean and kernel functions) are $\boldsymbol{\theta} = \{\beta,\boldsymbol{\alpha},\sigma_f,l,\sigma\}$. Keeping these values fixed, the joint distribution between the training data $D = \{\mathbf{x},\mathbf{y}\} = \{x_i, y_i\}_{i=1}^{N}$ and some previously unseen observations $\{\mathbf{x}_*,\mathbf{y}_*\} = \{\mathbf{x}_*[i],\mathbf{y}_*[i]\}_{i=1}^{M}$ (with additive noise) is multivariate Gaussian,

$$\begin{bmatrix}\mathbf{y}\\ \mathbf{y}_*\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\mathbf{m}\\ \mathbf{m}_*\end{bmatrix}, \begin{bmatrix}\mathbf{K}_{\mathbf{xx}}+\mathbf{R} & \mathbf{K}_{\mathbf{xx}_*}\\ \mathbf{K}_{\mathbf{x}_*\mathbf{x}} & \mathbf{K}_{\mathbf{x}_*\mathbf{x}_*}+\mathbf{R}_*\end{bmatrix}\right). \tag{8}$$

$$\mathbf{R} \triangleq \sigma^2\mathbf{I}_N$$
$$\mathbf{R}_* \triangleq \sigma^2\mathbf{I}_M \tag{9}$$

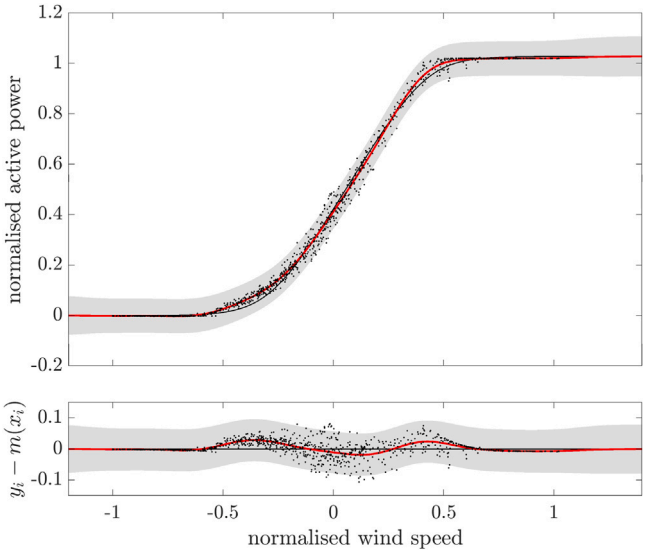

Fig. 5. Homoscedastic GP regression of the ideal power curve. The model in the data space (top) and the zero-mean transformed space (bottom). The black line shows the prior mean $\mathbf{m}$, the red line shows the predictive mean $\mathbb{E}[\mathbf{y}_*]$, and the shaded region shows three-sigma of the predictive variance $\mathbb{V}[\mathbf{y}_*]$.

where $\{\mathbf{R}, \mathbf{R}_*\}$ define the *noise kernels*, such that $\mathbf{I}_N$ denotes an $N \times N$ identity matrix, and $\mathbf{I}_M$ denotes an $M \times M$ identity matrix. Continuing similar notation, $\mathbf{m}_* = \left\{m\left(\mathbf{x}_*[i]\right)\right\}_{i=1}^{M}$ denotes the mean vector for the new observations.

According to the standard identity for conditioning a joint Gaussian distribution [16,27], the predictive equations can be defined,

$$
\begin{aligned}
p\left(\mathbf{y}_* \mid \mathbf{x}_*, D\right) &= \mathcal{N}\left(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*\right) \\
\boldsymbol{\mu}_* &\triangleq \mathbf{m}_* + \mathbf{K}_{\mathbf{x}_*\mathbf{x}}\left(\mathbf{K}_{\mathbf{xx}} + \mathbf{R}\right)^{-1}(\mathbf{y} - \mathbf{m}) \\
\boldsymbol{\Sigma}_* &\triangleq \mathbf{K}_{\mathbf{x}_*\mathbf{x}_*} - \mathbf{K}_{\mathbf{x}_*\mathbf{x}}\left(\mathbf{K}_{\mathbf{xx}} + \mathbf{R}\right)^{-1}\mathbf{K}_{\mathbf{x}\mathbf{x}_*} + \mathbf{R}_*
\end{aligned}
\tag{10}
$$

i.e. the mean of the posterior predictive function is $\mathbb{E}\left[\mathbf{y}_*\right] = \boldsymbol{\mu}_*$, and the variance about that mean is $\mathbb{V}\left[\mathbf{y}_*\right] = \mathrm{diag}(\boldsymbol{\Sigma}_*)$ (ignoring cross-terms).

Until this point, the hyperparameters $\boldsymbol{\theta} = \left\{\beta, \boldsymbol{\alpha}, \sigma_f, l, \sigma\right\}$ have been fixed. In practice, these are (typically) optimised through empirical Bayes [27], i.e. a type-II maximum likelihood [16], see Appendix A for details.

5.3. Heteroscedastic updates: Estimating input-dependent noise

Currently, the noise term ϵ_i in Eq. (1) has been governed by a single hyperparameter σ. When squared, σ defines the noise variance; in turn, this defines the noise kernel $\mathbf{R}$ (Eq. (9)). This setup enforces the assumption that the noise variance is *constant* across the input domain, leading to a *homoscedastic* GP — that is, the *noise variance* does not change over x_i. To demonstrate, Fig. 5 depicts the homoscedastic GP learnt from the ideal data (in Fig. 1). The model behaves as expected: the mean function of the prior approximates the relationship as far as possible, while the GP models the residual between this prior and the data. To highlight this effect, the residual modelled by the GP can be visualised in the zero-mean transformed space, that is $[y_i - m(x_i)]$, Fig. 5.

While the expected function $\mathbb{E}[\mathbf{y}_*]$ is representative of the general trend, the noise variance is poorly approximated when σ is constant. This is particularly apparent in the transformed space $\left(y_i - m\left(x_i\right)\right)$, where the noise (represented by the shaded area) is significantly overestimated at high/low wind speeds (towards the asymptotes) and underestimated in the near-linear (central) regions. In consequence, as proposed in [4], it is necessary to model power curve data with input-dependent noise, via *heteroscedastic* regression [29]. Specifically, the variance of the noise terms is now some function of the inputs x_i, such that,

$$\epsilon_i \sim \mathcal{N}(0, \sigma_i^2) \tag{11}$$

$$\sigma_i^2 = r(x_i) \tag{12}$$

The GP equations remain the same, other than (9), which defined a homoscedastic noise kernel. For a heteroscedastic process, the diagonal of the noise kernel is now defined by $r(x_i)$, rather than a constant, such that,

$$
\begin{aligned}
\mathbf{R} &\triangleq \mathrm{diag}\left(\left\{r(x_1), \ldots, r(x_N)\right\}\right) \\
\mathbf{R}_* &\triangleq \mathrm{diag}\left(\left\{r(x_{*1}), \ldots, r(x_{*M})\right\}\right)
\end{aligned}
\tag{13}
$$

where the off-diagonal elements are zero, $\mathbf{R}$ is an $N \times N$ matrix, and $\mathbf{R}_*$ is an $M \times M$ matrix.

Rather than specifying a functional form for the noise variance, an additional independent GP is used to infer the function $r(x_i)$. As σ must be strictly positive, the GP models the log-noise levels, denoted $g(x_i)$, such that,

$$\log(r(x_i)) = g(x_i) \sim GP(\mu_g, k_g(x_i, x_j)) \tag{14}$$

i.e. a GP prior with constant mean μ_g and a squared-exponential kernel. The kernel has the same form as Eq. (5), with a distinct length scale and process variance, such that the hyperparameters of the noise-process are $\boldsymbol{\zeta} = \{\mu_g, \sigma_g, l_g\}$.

The training points for the g-process can have arbitrary locations; in this case, it is convenient that they coincide with the f-process, such that $\mathbf{g} = \{g(x_i), \dots, g(x_N)\}$. Since the noise process has been introduced as additional (conditionally-independent) latent variables $\mathbf{g}$, the predictive distribution for $\mathbf{y}_*$ (previously Eq. (10)) is extended to [23],

$$p\left(\mathbf{y}_* \mid \mathbf{x}_*, \mathcal{D}\right) = \iint p\left(\mathbf{y}_* \mid \mathbf{x}_*, \mathbf{g}, \mathbf{g}_*, \mathcal{D}\right) p(\mathbf{g}, \mathbf{g}_* \mid \mathbf{x}_*, \mathcal{D}) d\mathbf{g} d\mathbf{g}_* \tag{15}$$

Fixing $\{\mathbf{g}, \mathbf{g}_*\}$, the predictive distribution $p\left(\mathbf{y}_* \mid \mathbf{x}_*, \mathbf{g}, \mathbf{g}_*, \mathcal{D}\right)$ is the same as before — with Eq. (13) defining the noise kernel $\mathbf{R}$.

Unfortunately, the term $p(\mathbf{g}, \mathbf{g}_* \mid \mathbf{x}_*, \mathcal{D})$ is problematic, as the integral is intractable. Various approximations of the integral can be implemented; including Monte-Carlo approximations, as well as variational inference [4,30,31]. A simple (and computationally-inexpensive) point-wise approximation of $\mathbf{g}$ is utilised here. This approach is convenient, since input-dependent noise can be implemented as an *update* step following inference of the OMGP (outlined in Section 6). In this case, the approximation was found to be representative of input-dependent noise for the power curve data.

Specifically, according to Kersting et al. [23], the *most likely* estimate of the target noise levels is assumed for the g-process, such that,

$$p\left(\mathbf{y}_* \mid \mathbf{x}_*, \mathcal{D}\right) \approx p\left(\mathbf{y}_* \mid \mathbf{x}_*, \hat{\mathbf{g}}, \hat{\mathbf{g}}_*, \mathcal{D}\right) \tag{16}$$

$$\{\hat{\mathbf{g}}, \hat{\mathbf{g}}_*\} \triangleq \text{argmax}_{\{\hat{\mathbf{g}}, \hat{\mathbf{g}}_*\}} \left\{p(\mathbf{g}, \mathbf{g}_* \mid \mathbf{x}_*, \mathcal{D})\right\} \tag{17}$$

i.e. *most* (all) of the density of $p(\mathbf{g}, \mathbf{g}_* \mid \mathbf{x}_*, \mathcal{D})$ is assumed to be concentrated around the mode $\{\hat{\mathbf{g}}, \hat{\mathbf{g}}_*\}$ [23].

5.3.1. Optimisation of the noise process

To obtain point-wise estimates of $\mathbf{g}$, a homoscedastic process is initially learnt by type-II ML – denoted G_1 – with hyperparameters $\boldsymbol{\theta}$. ($\mathbf{R}$ is a constant noise kernel, as in Eq. (9).) Given G_1, an empirical estimate of the most likely noise variance can be calculated for each training observation $\left\{x_i, y_i\right\} \in \mathcal{D}$, by considering a sample $\tilde{y}_i^{(j)}$ from the predictive distribution of G_1. If y_i and $\tilde{y}_i^{(j)}$ are viewed as two independent observations from the same underlying distribution, their arithmetic mean $0.5\left(y_i - \tilde{y}_i^{(j)}\right)^2$ is shown to be a valid approximation of the noise variance at x_i [23]. This can be improved by taking an expectation w.r.t. the predictive distribution, such that [23],

$$\log\left\{\mathbb{V}\left[y_i, G_1(x_i, \mathcal{D})\right]\right\} \approx g_i' \tag{18}$$

$$= \log\left\{\frac{1}{s}\sum_{j=1}^{s} 0.5\left(y_i - \tilde{y}_i^{(j)}\right)^2\right\} \tag{19}$$

here, s is the sample size from the predictive distribution of G_1. A suitably large value of s should lead to reasonable estimates: Kersting et al. [23] recommend $s \geq 100$, thus, in this case, $s = 100$. Having calculated $\mathbf{g}' = \{g_1', \dots, g_N'\}$, the noise process can be learnt – denoted G_2 – by type-II ML (given $\left\{\mathbf{g}', \mathbf{x}\right\}$) with distinct hyperparameters $\boldsymbol{\zeta}$. Then, conditioning a joint multivariate Gaussian (as before) the distribution $p(\mathbf{g}_* \mid \mathbf{x}_*, \mathbf{x}, \mathbf{g}')$ can be used to predict the (logarithmic) noise variance across the input space; in turn, defining $r(x_i)$.

The *heteroscedastic* GP – denoted G_3 – combines G_1 and G_2; i.e. G_2 models the input-dependent noise kernel according to Eq. (13) for the G_1 process. At this point, G_1 is set to G_3 ($G_1 \leftarrow G_3$) and each step is repeated until convergence in the marginal likelihood (of the heteroscedastic process G_3). The optimisation procedure is summarised in Appendix B. Learning $\mathbf{g}$ in this way effectively minimises the *average* distance between the target output y_i and the predictive distribution of the (heteroscedastic) process G_3 at the training inputs [23].

5.4. Heteroscedastic regression of the ideal power curve

The optimised heteroscedastic process for the ideal data is shown in Fig. 6. Unlike the homoscedastic example (Fig. 5) the model is representative of input-dependent noise; to highlight this, the lower sub-plots illustrate the changing variance (shaded regions) and associated noise-levels over the inputs (blue line). As expected, a lower variance is associated with the tails of the sigmoid and a larger variance at the centre. To quantify improvements, the joint-log-likelihood of the training and test data under the model can be monitored — this increases from 2.31×10^3 to 3.31×10^3, highlighting that input-dependent noise better approximates the variance in the data.

The results so far, however, have shown a regression of the ideal observations only, similar to [4]. The OMGP is now introduced to model curtailed data, such as those in Figs. 2 and 3.

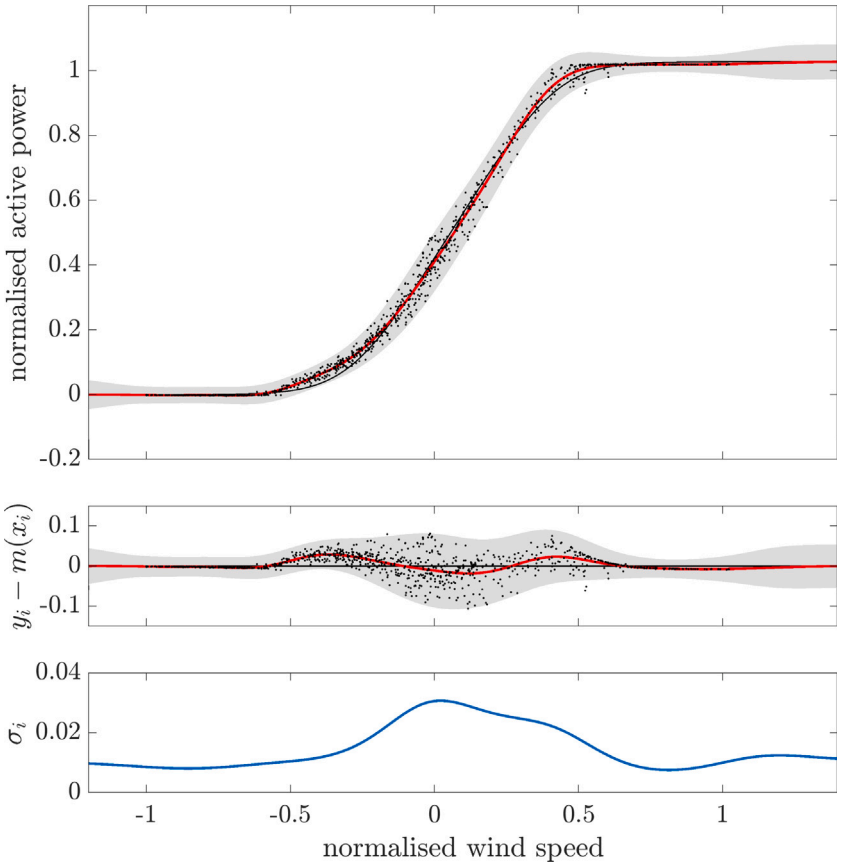

Fig. 6. Heteroscedastic GP regression of the ideal power curve. The model in the original space (top), the zero-mean transformed space (middle), and the expectation of the noise function $\sigma_i = \mathbb{E}\left[\sqrt{r(x_i)}\right]$ (bottom). The black line shows the prior mean $\mathbf{m}$, the red line shows the predictive mean $\mathbb{E}[\mathbf{y}_*] = \boldsymbol{\mu}_*$, and the shaded region shows three-sigma of the predictive variance $\mathbb{V}[\mathbf{y}_*] = \text{diag}(\boldsymbol{\Sigma}_*)$.

6. An overlapping mixture of Gaussian processes

The overlapping mixture of Gaussian processes (OMGP) model [15,32] is introduced to infer regression functions given curtailed power curve data. Here, the notation follows that of Lázaro-Gredilla et al. [15]. Rather than a single GP, the OMGP assumes multiple latent functions to describe the data, such that,

$$y_i^{(k)} = \left\{ f^{(k)}(x_i) + \epsilon_i \right\}_{k=1}^{K} \tag{20}$$

i.e. each observation is found by evaluating one of K latent functions, with additive noise: for now, each process is homoscedastic. As discussed, labels to assign observations to functions are unknown. In consequence, a latent variable is introduced to the model, $\mathbf{Z}$ – this is a binary indicator matrix, such that $\mathbf{Z}[i,k] \neq 0$ indicates that observation i was generated by function k. There is only one non-zero entry per row in $\mathbf{Z}$ (each observation is found by evaluating one function only).

The likelihood of the OMGP is, therefore [15],

$$p\left(\mathbf{y} \mid \left\{\mathbf{f}^{(k)}\right\}_{k=1}^{K}, \mathbf{Z}, \mathbf{x}\right) = \prod_{i,k=1}^{N,K} p\left(y_i \mid f^{(k)}(x_i)\right)^{\mathbf{Z}[i,k]} \tag{21}$$

As with the conventional GP, prior distributions are placed over the latent functions and variables,

$$P(\mathbf{Z}) = \prod_{i,k=1}^{N,K} \boldsymbol{\Pi}[i,k]^{\mathbf{Z}[i,k]} \tag{22}$$

$$f^{(k)}(x_i) \sim \mathcal{GP}\left(m^{(k)}\left(x_i\right), k^{(k)}\left(x_i, x_j\right)\right) \tag{23}$$

$$\epsilon_i^{(k)} \sim \mathcal{N}(0, \sigma^2) \tag{24}$$

where Eq. (22) is the prior over the indicator matrix, such that $\boldsymbol{\Pi}[i,:]$ is a histogram over the K components for the ith observation, and $\sum_{k=1}^{K} \boldsymbol{\Pi}[i,k] = 1$. (Note, colon notation is used to index all columns or rows in a matrix.) The terms in Eq. (23) are independent

GP priors over each latent function $f^{(k)}$ with distinct mean/kernel functions $\left(m^{(k)}\left(x_i\right), k^{(k)}\left(x_i, x_j\right)\right)$. To reduce the number of latent variables, the prior over the noise variances is defined by a *shared* hyperparameter σ (this is modified later, in the heteroscedastic updates).

The collected hyperparameters for the model are $\left\{\left\{\boldsymbol{\theta}_k\right\}_{k=1}^{K}, \boldsymbol{\Pi}\right\}$. The notation $\boldsymbol{\theta}_k$ denotes a distinct set of mean/kernel function hyperparameters for the kth component (including the noise kernel). Referring back to the curtailed data in Fig. 2, it is now possible to encode prior engineering knowledge of the expected functions through the covariance, mean, and hyperparameters. Here, it is argued that the following are known, given prior knowledge of wind turbine power curves:

- given the training data (and possibly prior knowledge of the operational conditions) it should be clear that three latent functions will be representative of the data, such that[3] $K = 3$;
- for the zero-power relationship, a linear regression (with a constant kernel) should be representative;
- for the remaining functions (ideal and curtailed data) the soft-clip Eq. (6) appropriately describes the expected relationships.

In this setting, while $K = 3$, the prior includes two independent GPs with a soft-clip mean (Eq. (6)) and squared-exponential kernel (Eq. (5)) function. These priors correspond to the ideal and curtailed curves. For the final component, a constant kernel is selected $k^{(3)}(x_i, x_j) = c$; this reduces the latent function to a (zero-gradient) linear regression, to approximate the zero-power data. To summarise, the hyperparameters (of the prior) of the model are: $\boldsymbol{\theta}_k = \{\beta_k, \boldsymbol{\alpha}_k, \sigma_f^{(k)}, l_k, \sigma\}_{k=1}^{2}$ and $\boldsymbol{\theta}_3 = \{c, \sigma\}$.

6.1. A note on model assumptions

It is important to clarify the modelling assumptions. While the OGMP infers labels $\mathbf{Z}$, to associate measurements to functions in an unsupervised manner, the number of functional components (K) and their priors must be defined in advance. (This concept is somewhat analogous to unsupervised learning with Gaussian Mixture models [27].) As such, while an engineer is not required to label the data, they are required to predefine an appropriate number of functions. Here, it is assumed that this can be determined by inspecting the static training-set (i.e. Fig. 2) in an offline sense. In certain scenarios, however, predefining K and the prior distributions is less trivial; incremental/online learning, for example. There are several options in this setting. One can select an appropriate number of components via cross-validation, considering quantities such as the Bayesian Information Criterion (BIC) or Bayes factors [27] – an example of cross-validation is provided in Section 7.1 and Appendix D. Alternatively, K could be considered as an additional latent variable, and its estimation could be included in the inference. Unsurprisingly inferring K in this way is more involved, as presented by Ross and Dy [33].

It is reiterated that the training-data consider a subset of possible curtailments (described in Section 4.2). This consideration should not be an issue in practice, as the model is flexible in the power curves it can represent. To demonstrate, the OMGP is learnt for another set of curtailments, measured from four *alternative* turbines in the wind farm — the results are presented in Section 7. It should be acknowledged, of course, that inference will slow down as more data (or components) are included – a typical caveat when learning from data. In the context of Gaussian processes, there are a number of options; for example, sparse approximations could be explored [34].

6.2. Variational approximation

Due to the latent variables $\left\{\mathbf{f}^{(k)}\right\}$ and $\mathbf{Z}$, computation of the posterior $p\left(\left\{\mathbf{f}^{(k)}\right\}_{k=1}^{K}, \mathbf{Z} \mid \mathbf{x}, \mathbf{y}\right)$ is now intractable; thus, variational inference (VI) [31] is implemented. Specifically, VI involves specifying an approximate density family $q(\mathbf{a}) \in \mathcal{Q}$ over the target conditional $p(\mathbf{a} \mid \mathbf{b})$. The best candidate $\hat{q}(\mathbf{a})$ can be viewed as $q(\mathbf{a}) \in \mathcal{Q}$ that is *closest* to the (unknown) target $p(\mathbf{a} \mid \mathbf{b})$ in terms of the KL-divergence,

$$\hat{q}(\mathbf{a}) = \operatorname{argmin}_{q(\mathbf{a}) \in \mathcal{Q}} \{KL\left(q(\mathbf{a}) \,||\, p(\mathbf{a} \mid \mathbf{b})\right)\} \tag{25}$$

Once found, $\hat{q}(\mathbf{a})$ is the best approximation of $p(\mathbf{a} \mid \mathbf{b})$ within the family $\mathcal{Q}$ [31]. (In this case, $\mathbf{a} \triangleq \left\{\left\{\mathbf{f}^{(k)}\right\}, \mathbf{Z}\right\}$ and $\mathbf{b} \triangleq \{\mathbf{y}\}$.) The KL divergence for Eq. (25) is defined,

$$KL\left(q(\mathbf{a}) \,||\, p(\mathbf{a} \mid \mathbf{b})\right) = \mathbb{E}_{q(\mathbf{a})}[\log q(\mathbf{a})] - \mathbb{E}_{q(\mathbf{a})}[\log p(\mathbf{a} \mid \mathbf{b})] \tag{26}$$

$$= \mathbb{E}_{q(\mathbf{a})}[\log q(\mathbf{a})] - \mathbb{E}_{q(\mathbf{a})}[\log p(\mathbf{a}, \mathbf{b})] + \log p(\mathbf{b}) \tag{27}$$

(Steps (26) to (27) uses log rules while expanding the conditional.) Eq. (27) reveals the dependence on $p(\mathbf{b})$, which is intractable, and why VI is needed in the first place [31]. Therefore, rather than the KL divergence (27), an alternative object is optimised that is equivalent to the (negative) KL divergence up to the term $\log p(\mathbf{b})$, which is a constant with respect to $q(\mathbf{z})$; that is,

$$\mathcal{L}_b(\mathbf{a}) = \mathbb{E}_{q(\mathbf{a})}[\log p(\mathbf{a}, \mathbf{b})] - \mathbb{E}_{q(\mathbf{a})}[\log q(\mathbf{a})] \tag{28}$$

$$= \int q(\mathbf{a}) \log \frac{p(\mathbf{a}, \mathbf{b})}{q(\mathbf{a})} \, \mathrm{d}\,\mathbf{a} \tag{29}$$

[3] While it is assumed $K = 3$, the point-wise classification of each datum remains unknown.

This quantity is the referred to as the *evidence lower bound* (elbo). From (27), it can be seen that *maximising* this object will *minimise* the KL divergence between $q(\mathbf{a})$ and $p(\mathbf{a} \mid \mathbf{b})$.

Conveniently, Eq. (29) can be used to construct a lower bound on the marginal likelihood $p(\mathbf{b})$: i.e. rearranging Eq. (27) and substituting in (28) leads to,

$$\log p(\mathbf{b}) = KL\left(q\left(\mathbf{a}\right) \parallel p\left(\mathbf{a} \mid \mathbf{b}\right)\right) + \mathcal{L}_b \tag{30}$$

Since $KL(\cdot) \geq 0$ [35], it follows that the evidence is lower-bounded by the elbo, in other words $\log p(\mathbf{b}) \geq \mathcal{L}_b$. This inequality is useful, as $\mathcal{L}_b$ can be used to monitor the marginal likelihood during inference/optimisation (as with the conventional GP, Eq. (A.3)). Substituting notation $\mathbf{a} \triangleq \left\{\left\{\mathbf{f}^{(k)}\right\}, \mathbf{Z}\right\}$ and $\mathbf{b} \triangleq \{\mathbf{y}\}$ in (29), leads to (32),

$$\log p(\mathbf{y} \mid \mathbf{x}) = \log \iint p\left(\left\{\mathbf{f}^{(k)}\right\}, \mathbf{Z}, \mathbf{y}, \mathbf{x}\right) p\left(\left\{\mathbf{f}^{(k)}\right\}\right) p(\mathbf{Z}) \, \mathrm{d}\left\{\mathbf{f}^{(k)}\right\} \mathrm{d}\mathbf{Z} \tag{31}$$

$$\geq \mathcal{L}_b = \iint q\left(\left\{\mathbf{f}^{(k)}\right\}, \mathbf{Z}\right) \log \frac{p\left(\left\{\mathbf{f}^{(k)}\right\}, \mathbf{Z}, \mathbf{y}, \mathbf{x}\right)}{q\left(\left\{\mathbf{f}^{(k)}\right\}, \mathbf{Z}\right)} \, \mathrm{d}\left\{\mathbf{f}^{(k)}\right\} \mathrm{d}\mathbf{Z}$$

$$= \iint q\left(\left\{\mathbf{f}^{(k)}\right\}, \mathbf{Z}\right) \log \frac{p\left(\mathbf{y} \mid \left\{\mathbf{f}^{(k)}\right\}, \mathbf{Z}, \mathbf{x}\right) p(\mathbf{Z}) \prod_{k=1}^{K} p\left(\mathbf{f}^{(k)} \mid \mathbf{x}\right)}{q\left(\left\{\mathbf{f}^{(k)}\right\}, \mathbf{Z}\right)} \, \mathrm{d}\left\{\mathbf{f}^{(k)}\right\} \mathrm{d}\mathbf{Z} \tag{32}$$

A family of variational distributions $q \in \mathcal{Q}$ is now chosen to approximate $p\left(\left\{\mathbf{f}^{(k)}\right\}, \mathbf{Z} \mid \mathbf{x}, \mathbf{y}\right)$ such that a mean field assumption is implemented: i.e. q factorises, $q\left(\left\{\mathbf{f}^{(k)}\right\}, \mathbf{Z}\right) = q\left(\left\{\mathbf{f}^{(k)}\right\}\right) q(\mathbf{Z})$. In consequence, due to conjugacy, it is possible to analytically update each latent variable in turn (while keeping the others fixed) such that the bound $\mathcal{L}_b$ is maximised (with respect to that variable). Updates for each factor are iterated until convergence in the lower bound $\mathcal{L}_b$.[4]

6.2.1. Mean-field updates

Firstly, assuming $q\left(\left\{\mathbf{f}^{(k)}\right\}\right)$ is known – and therefore the marginals for each component $q\left(\mathbf{f}^{(k)}\right) = \mathcal{N}\left(\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)}\right)$ – it is possible to analytically maximise $\mathcal{L}_b$ with respect to $q(\mathbf{Z})$, by setting the derivative of the bound to zero, and constraining q to be a probability density [15],

$$q(\mathbf{Z}) = \prod_{i=1,k=1}^{N,K} \hat{\boldsymbol{\Pi}}[i,k]^{\mathbf{Z}[i,k]}, \quad \text{s.t.} \quad \hat{\boldsymbol{\Pi}}[i,k] \propto \boldsymbol{\Pi}[i,k] \exp\left(a_{ik}\right) \tag{33}$$

$$a_{ik} \triangleq -\frac{1}{2\sigma^2}\left(\left(y_i - \boldsymbol{\mu}_i^{(k)}\right)^2 + \boldsymbol{\Sigma}^{(k)}[i,i]\right) - \frac{1}{2}\log\left(2\pi\sigma^2\right)$$

where Eq. (33) implies the approximated distribution $q(\mathbf{Z})$ is factorised for each sample [15].

Conversely, assuming $q(\mathbf{Z})$ is known, $\mathcal{L}_b$ can maximised with respect to each latent function $q\left(\left\{\mathbf{f}^{(k)}\right\}\right)$,

$$q\left(\mathbf{f}^{(k)}\right) = \mathcal{N}\left(\mathbf{f}^{(k)} \mid \boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)}\right) \tag{34}$$

$$\boldsymbol{\Sigma}^{(k)} \triangleq \left(\mathbf{K}_{\mathbf{xx}}^{-1(k)} + \mathbf{B}^{(k)}\right)^{-1}$$

$$\boldsymbol{\mu}^{(k)} \triangleq \mathbf{m}^{(k)} + \boldsymbol{\Sigma}^{(k)}\mathbf{B}^{(k)}\left(\mathbf{y} - \mathbf{m}^{(k)}\right)$$

where $\mathbf{B}^{(k)}$ is a $N \times N$ diagonal matrix (off-diagonals are zero) with elements,

$$\mathbf{B}^{(k)} = \mathrm{diag}\left(\left\{\frac{\hat{\boldsymbol{\Pi}}[1,k]}{\sigma^2}, \ \ldots \ , \frac{\hat{\boldsymbol{\Pi}}[N,k]}{\sigma^2}\right\}\right) \tag{35}$$

To find a candidate $\hat{q}$ that is closest to the true posterior, $q(\mathbf{Z})$ and $q\left(\left\{\mathbf{f}^{(k)}\right\}\right)$ are initialised from their priors, and they are iteratively updated by alternating Eqs. (33) and (34). Both updates are optimal with respect to the distribution that they compute; therefore, they are guaranteed to increase the (lower bound) on the log-marginal-likelihood [15], Eq. (32).

6.2.2. Monitoring convergence: An improved lower bound

As in [30], an improved bound is used to *monitor* convergence, introduced by King and Lawrence [36]. This object, denoted $\mathcal{L}_{bc}$, also lower-bounds the marginal likelihood, while being an upper-bound on the standard variational bound $\mathcal{L}_b$ (Eq. (32)). (That is, if $\mathcal{L}_b$ is subtracted from the improved bound, the result is a KL divergence — as $KL(\cdot) \geq 0$, this implies that $\mathcal{L}_{bc}$ upper-bounds $\mathcal{L}_b$.) The bound can be defined when the term $\log \int p\left(\left\{\mathbf{f}^{(k)}\right\}, \mathbf{Z}, \mathbf{y}, \mathbf{x}\right) p(\mathbf{Z}) \, \mathrm{d}\mathbf{Z}$ – in the true marginal likelihood, Eq. (31) – is replaced with $\int q(\mathbf{Z}) \log \frac{p\left(\left\{\mathbf{f}^{(k)}\right\}, \mathbf{Z}, \mathbf{y}, \mathbf{x}\right) p(\mathbf{Z})}{q(\mathbf{Z})} \, \mathrm{d}\mathbf{Z}$. Following substitution, it is possible to integrate out $p\left(\left\{\mathbf{f}^{(k)}\right\}\right)$ analytically. Alternatively, Lázaro-Gredilla and Titsias [30] show that it is possible to obtain the corrected bound by optimally removing $p\left(\left\{\mathbf{f}^{(k)}\right\}\right)$ from the standard bound. The (implementation friendly) expression for $\mathcal{L}_{bc}$ is as follows [30],

$$\log p(\mathbf{y} \mid \mathbf{x}) \geq \mathcal{L}_{bc}$$

[4] At this stage in the inference, the hyperparameters of the model $\left\{\{\boldsymbol{\theta}_k\}_{k=1}^{K}, \boldsymbol{\Pi}\right\}$ are fixed — they will be optimised later.

$$= \sum_{k=1}^{K} \left(-\frac{1}{2} \left\| \mathbf{R}^{(k)\mathsf{T}} \backslash \left(\mathbf{B}^{(k)\frac{1}{2}} \left(\mathbf{y} - \mathbf{m}^{(k)} \right) \right) \right\|^2 - \sum_{i=1}^{N} \log \mathbf{R}^{(k)}[i,i] \right) \dots$$

$$- \mathrm{KL}\left(q(\mathbf{Z}) \,\middle\|\, p(\mathbf{Z}) \right) - \frac{1}{2} \sum_{i=1,k=1}^{N,K} \hat{\boldsymbol{\Pi}}[i,k] \log\left(2\pi\sigma^2 \right) \tag{36}$$

$$\mathbf{R}^{(k)} \triangleq \mathrm{chol}\left(\mathbf{I} + \mathbf{B}^{(k)\frac{1}{2}} \mathbf{K}_{\mathbf{xx}}^{(k)} \mathbf{B}^{(k)\frac{1}{2}} \right)$$

$$\mathrm{KL}\left(q(\mathbf{Z}) \,\middle\|\, p(\mathbf{Z}) \right) \triangleq \sum_{i=1,k=1}^{K,N} \hat{\boldsymbol{\Pi}}[i,k] \log \frac{\boldsymbol{\Pi}[i,k]}{\hat{\boldsymbol{\Pi}}[i,k]}$$

where chol(·) is the Cholesky decomposition and the backslash operator $\mathbf{A}\backslash\mathbf{B}$ solves the systems of linear equations $\mathbf{Ac} = \mathbf{B}$ for $\mathbf{c}$. The improved, tighter bound is independent of $p\left(\left\{\mathbf{f}^{(k)}\right\}\right)$ – hence it can be referred to as the *marginalised variational bound* [30]. In words, this implies that $\mathcal{L}_b$ is the same as $\mathcal{L}_{bc}$ – for a given $q(\mathbf{Z})$ – when an optimal choice for $p\left(\left\{\mathbf{f}^{(k)}\right\}\right)$ is made [15]. In consequence, the bound is more stable over different hyperparameter values [36], and it is more efficient when optimising $\left\{ \left\{\boldsymbol{\theta}_k\right\}_{k=1}^{K}, \boldsymbol{\Pi} \right\}$ through type-II ML (the optimisation scheme is outlined below).

6.2.3. Optimisation of hyperparameters

A variational inference and Expectation Maximisation (EM) scheme is implemented [31]. The strategy iteratively updates the *approximate* (factorised) posterior and then optimises the hyperparameters of the model, while the (improved) lower bound $\mathcal{L}_{bc}$ on the marginal likelihood is maximised. The EM steps are repeated until convergence:

1. E-step: mean field updates — iterate Eqs. (33) and (34) until convergence in $\mathcal{L}_{bc}$ (or $\mathcal{L}_b$), hyperparameters are fixed.
2. M-step: optimise the lower bound $\mathcal{L}_{bc}$ w.r.t. all hyperparameters until convergence,

$$\left\{ \left\{\hat{\boldsymbol{\theta}}_k\right\}_{k=1}^{K}, \hat{\boldsymbol{\Pi}} \right\} = \underset{\left\{ \left\{\boldsymbol{\theta}_k\right\}_{k=1}^{K}, \boldsymbol{\Pi} \right\}}{\mathrm{argmax}} \left\{ \mathcal{L}_{bc} \right\}$$

   the distribution $q(\mathbf{Z})$ is kept fixed.

Having initialised each component from the prior, steps 1 and 2 are iterated until convergence in $\mathcal{L}_{bc}$ (of the M-step).

6.2.4. Predictive equations

Having learnt the OMGP, it can be used to estimate the latent variables and functions. These predications are critical in the context of performance monitoring: i.e. for a given measurement of wind speed x_i, the OMGP can predict the power output y_i, and classify the trend (or curtailment) $k \in \{1, \dots, K\}$. The posterior predictive likelihood given the unseen inputs $\mathbf{x}_*$ is,

$$p(\mathbf{y}_* \mid \mathbf{x}_*, D) \approx \sum_{k=1}^{K} \boldsymbol{\Pi}[*,k] \int p\left(\mathbf{y}_* \mid \mathbf{f}^{(k)}, \mathbf{x}_*, D\right) q\left(\mathbf{f}^{(k)} \mid D\right) \mathrm{d}\,\mathbf{f}^{(k)} \tag{37}$$

$$= \sum_{k=1}^{K} \boldsymbol{\Pi}[*,k]\, \mathcal{N}\left(\mathbf{y}_* \mid \boldsymbol{\mu}_*^{(k)}, \boldsymbol{\Sigma}_*^{(k)}\right) \tag{38}$$

$$\boldsymbol{\mu}_*^{(k)} \triangleq \mathbf{m}_*^{(k)} + \mathbf{K}_{\mathbf{x}_*\mathbf{x}}^{(k)} \left(\mathbf{K}_{\mathbf{xx}}^{(k)} + \mathbf{B}^{(k)-1} \right)^{-1} \left(\mathbf{y} - \mathbf{m}^{(k)} \right)$$

$$\boldsymbol{\Sigma}_*^{(k)} \triangleq \mathbf{K}_{\mathbf{x}_*\mathbf{x}_*}^{(k)} - \mathbf{K}_{\mathbf{x}_*\mathbf{x}}^{(k)} \left(\mathbf{K}_{\mathbf{xx}}^{(k)} + \mathbf{B}^{(k)-1} \right)^{-1} \mathbf{K}_{\mathbf{xx}_*}^{(k)} + \mathbf{R}_*^{(k)}$$

$$\mathbf{R}_*^{(k)} \triangleq \sigma^2 \mathbf{I}_M \tag{39}$$

The prior mixing proportion for new observations $\boldsymbol{\Pi}[*,k]$ is a fixed hyperparameter, weighting each component equally *a priori*, such that $\boldsymbol{\Pi}[*,k] = 1/K$. Interestingly, the predictive equations for the OMGP are similar to the conventional GP (Eq. (10)), however, the noise component for the training data ($\mathbf{B}^{(k)-1}$) is scaled according to $\hat{\boldsymbol{\Pi}}[i,k]^{-1}$ [15]. Thus, the noise component effectively *weights* the contribution of each observation in D to its posterior predictive component in the mixture.

Another useful prediction categorises observations according to the most likely component k. For the training data D, this is simply the *maximum a posteriori* (MAP) estimate, given the approximated posterior (33),

$$\hat{k}_i = \underset{k}{\mathrm{argmax}} \left\{ \hat{\boldsymbol{\Pi}}[i,k] \right\} \tag{40}$$

For the test-data (i.e. weekly wind-power data $\{\mathbf{x}_*, \mathbf{y}_*\}$) the posterior predictive class component k_* is,

$$p(k_* \mid \mathbf{x}_*, \mathbf{y}_*, D) = \frac{p(\mathbf{y}_* \mid \mathbf{x}_*, k, D)\, \boldsymbol{\Pi}[*,k]}{p(\mathbf{y}_* \mid \mathbf{x}_*, D)} \tag{41}$$

where the denominator (evidence) was defined in (38), and the numerator is,

$$p(\mathbf{y}_* \mid \mathbf{x}_*, k_*, D)\, p(k_*) \triangleq \mathcal{N}\left(\mathbf{y}_* \mid \boldsymbol{\mu}_*^{(k)}, \boldsymbol{\Sigma}_*^{(k)}\right) \boldsymbol{\Pi}[*,k] \tag{42}$$

the MAP class component $\hat{k}_*$ can then be defined,

$$\hat{k}_* = \underset{k_*}{\operatorname{argmax}} \left\{ p(k_* \mid \mathbf{x}_*, \mathbf{y}_*, \mathcal{D}) \right\} \tag{43}$$

Note, classifying new data according to $\hat{k}_*$ is only possible when *both* $\mathbf{x}_*$ and $\mathbf{y}_*$ have been observed. This implies that predictions using Eq. (43) should be used in certain monitoring applications (as demonstrated in the results).

6.2.5. Input dependent noise approximations for the OMGP

At this stage, it is possible to apply heteroscedastic updates to the OMGP, according to the method in Section 5.3. In this case, for each kth component, the noise variance is now considered a function of the inputs,

$$\sigma_i^{(k)^2} = r^{(k)}(x_i) \tag{44}$$

$$\log r^{(k)}(x_i) = g^{(k)}(x_i) \sim GP(\mu_g^{(k)}, k_g^{(k)}(x_i, x_j)) \tag{45}$$

i.e. there are K GPs (with hyperparameters $\boldsymbol{\zeta}_k = \{\mu_g^{(k)}, \sigma_g^{(k)}, l_g^{(k)}\}$) to describe input-dependent noise for each function in the mixture — rather than a single, shared hyperparameter σ.

Again, the predictive Eq. (38) remains similar, where the noise kernels are updated. In this case, $\mathbf{B}^{(k)}$ (from Eq. (35)) becomes,

$$\mathbf{B}^{(k)} = \operatorname{diag}\left(\left\{ \frac{\hat{\boldsymbol{\Pi}}[\mathcal{I}_1^{(k)}, k]}{r^{(k)}\left(x_{\mathcal{I}_1^{(k)}}\right)}, \ \dots \ , \frac{\hat{\boldsymbol{\Pi}}[\mathcal{I}_N^{(k)}, k]}{r^{(k)}\left(x_{\mathcal{I}_N^{(k)}}\right)} \right\} \right) \tag{46}$$

where the indices $\boldsymbol{\mathcal{I}}_k = \{\mathcal{I}_1^{(k)}, \dots \mathcal{I}_N^{(k)}\}$ correspond to observations in $\mathcal{D}$ whose MAP label is k. Formally, $\{x_i, y_i\}_{i \in \boldsymbol{\mathcal{I}}_k} \in \mathcal{D}$, where $\hat{k}_{i \in \boldsymbol{\mathcal{I}}_k} = k$. Additionally, $\mathbf{R}_*^{(k)}$ from Eq. (39) is updated,

$$\mathbf{R}_*^{(k)} \triangleq \operatorname{diag}\left(\left\{ r^{(k)}(x_{*1}), \dots, r^{(k)}(x_{*M}) \right\} \right) \tag{47}$$

In summary, to approximate the noise-process for each component, the training data are split into K subsets, according to the MAP classification (43) given the homoscedastic OMGP and the training data. That is, the noise-processes are approximated for each component, given the training data that are associated with that component (according to $\hat{k}_i$) and the framework outlined in Section 5.3.

7. Results

In total, 8900 observations were sampled from the wind farm data, corresponding to a (selected) subset of seven operational turbines over nine weeks. As aforementioned, three trends are present in these data; additional curtailments may be observed in practical data, and can be included in the OMGP if necessary — an alternative example is provided in Section 7.1. The data are shown in Figs. 2 and 3. Approximately 1/3 ($N = 2980$ observations) were using for training here, and the remaining data ($M = 5920$ observations) were used as an independent test-set.

OMGP regression of the curtailed data is shown in Fig. 7. Given the training observations (larger • markers), the model has inferred the multivalued behaviour in an unsupervised manner, including the ideal curve (orange), ≈ 50% curtailment (green), and the zero-power behaviour (purple).

Visually, the model is representative of the underlying functions, and it appears to generalise to the test data (smaller · markers). Importantly, the GP successfully models the residual between prior engineering knowledge (encoded in the parametrised mean, shown by the black lines in Fig. 7) and the data. Generally, the heteroscedastic updates are representative. The noise levels are (perhaps) overestimated towards the asymptotes of the power curves (high and low wind speeds). Additionally, the noise for the zero-power trend (purple) is overestimated, as it captures some of the data associated with the ideal/curtailed data — around negative 0.5 normalised wind speed. Smaller length scales $l_g^{(k)}$ in the noise-processes $g^{(k)}$ might prove appropriate, as there is no guarantee that the parameter set $\left\{ \{\theta_k, \boldsymbol{\zeta}_k\}_{k=1}^K, \boldsymbol{\Pi} \right\}$ represents the *global* minimum of the log-marginal-likelihood. However, following several initialisations, this realisation was the most representative (and repeatable).

To quantify performance, the normalised mean squared-error (NMSE) and Mahalanobis squared-distance (MSD) are provided. As the OMGP is a mixture, each test observation is assessed against its most likely component $\hat{k}_*$. In other words, NMSE and (normalised) MSD are assessed for each function with respect to their most likely data — the corresponding subsets are shown for each component in the (lower) plots of Fig. 7.

$$N\hat{M}SE = \frac{100}{M\sigma_{\mathbf{y}_*}^2} \left(\boldsymbol{\mu}_*^{(\hat{k}_*)} - \mathbf{y}_* \right)^\top \left(\boldsymbol{\mu}_*^{(\hat{k}_*)} - \mathbf{y}_* \right) \tag{48}$$

Similarly, the MSD is,

$$M\hat{S}D = \frac{1}{M} \sum_{i=1}^{M} \frac{\left(\boldsymbol{\mu}_*^{(\hat{k}_*)}[i] - \mathbf{y}_*[i] \right)^2}{\boldsymbol{\Sigma}^{(\hat{k}_*)}[i,i]} \tag{49}$$

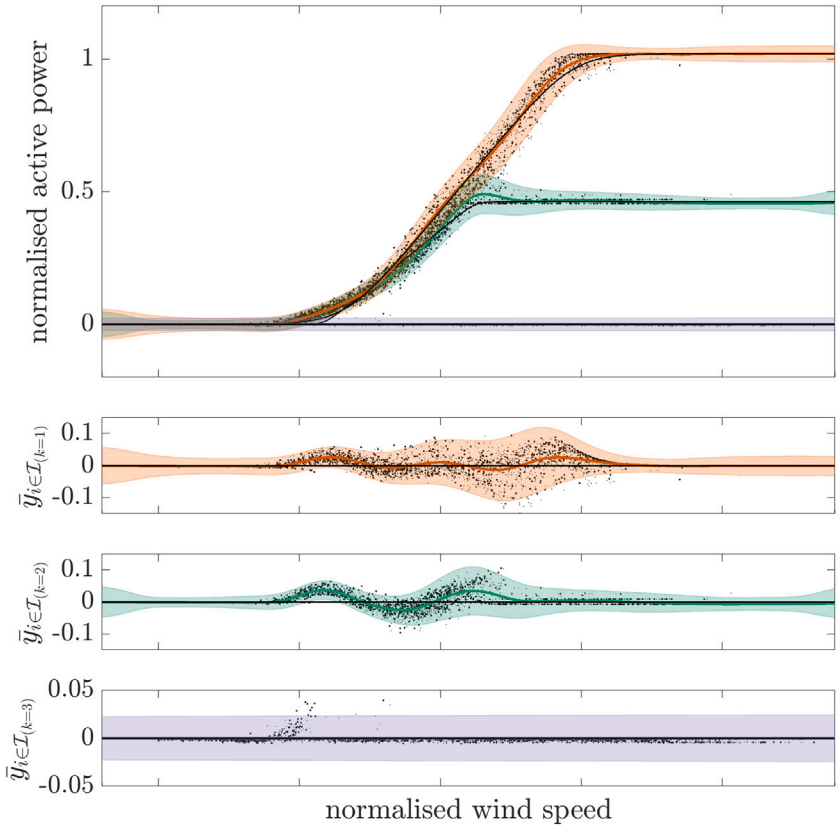

Fig. 7. Heteroscedastic OMGP regression of curtailed power curve data. The mixture model in the original space (top), and each component in the zero-mean transformed space, i.e. $\bar{y}_i = y_i - m^{(k)}(x_i)$ (bottom three plots). Black lines show the mean functions of the prior $\mathbf{m}^{(k)}$. The green, orange, and purple lines show the predictive mean $\boldsymbol{\mu}_*^{(k)}$, and shaded regions show three-sigma of the predictive variance $\mathrm{diag}(\boldsymbol{\Sigma}_*^{(k)})$. Small · markers show the test set, and larger • markers show the training set. For each component, the data correspond to their MAP function, according to $\hat{k}_i$ and $\hat{k}_*$.

Table 1
Model performance metrics for the curtailed power curve data.

	Conventional regression		Mixture of regressions	
	RVM	GP	OMGP	Het-OMGP
NMSE	47.13	46.97	0.26	0.26
MSD	1.01	1.02	1.00	0.73

Table 1 quantifies significant improvements in representing the curtailed power data with a heteroscedastic OMGP. For reference, an alternative probabilistic regression is included, previously applied in the literature [37], the Relevance Vector Machine (RVM); implementation details are provided in Appendix C. It is reiterated, however: the focus is to show improvements of a mixture of regressions, rather than improvements between conventional regression models.

The NMSE shows a marked advantage in representing the data with multiple latent functions. Nonetheless, the NMSE does not highlight advantages of heteroscedastic updates, since the metric (48) does not consider the predictive variance $\boldsymbol{\Sigma}_*^{(k)}$. Therefore, the (normalised) MSD in Table 1 highlights improvements when modelling input-dependent noise for the mixture[5].

As discussed, certain hyperparameters can be interpreted. $\alpha_1^{(k)}$ corresponds to the maximum (normalised) power in the prior, and β_k determines the rate of convergence (of the asymptote) for priors with sigmoidal mean functions. As expected, for the ideal curve ($k = 1$) the mean of the prior tends to $\alpha_1^{(1)} = 1.021$. For $k = 2$ the asymptote tends to $\alpha_1^{(2)} = 0.4623$; this provides a more accurate estimate of the maximum curtailed output (46.23% rather than $\approx 50\%$). As expected, the rate of convergence is greater for the curtailment data ($\beta_2 = 28.8$) and lower for the ideal data ($\beta_1 = 11.4$); this can be visualised in the plots of the prior mean functions (black lines) Fig. 7.

[5] It is acknowledged that the MSD is less useful when assessing the *fit* of the OMGP, as the *error* is scaled by the predictive variance $\boldsymbol{\Sigma}_*^{(k)}$; thus, the MSD is used only to assess the predictive variance $\boldsymbol{\Sigma}_*^{(k)}$.

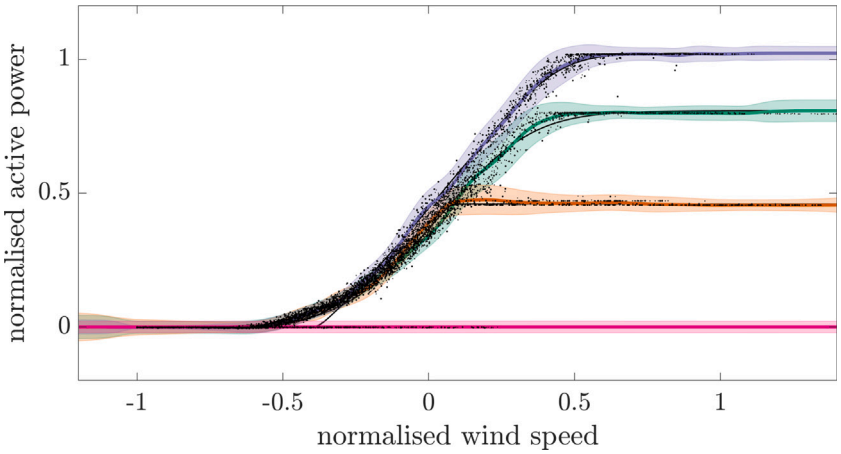

Fig. 8. Heteroscedastic OMGP of curtailed data from an alternative group of turbines, also exhibiting 80% curtailment. Black lines show the mean functions of the prior $\mathbf{m}^{(k)}$. The green, orange, purple, and pink lines show the predictive mean $\boldsymbol{\mu}_*^{(k)}$, and shaded regions show three-sigma of the predictive variance $\mathrm{diag}(\boldsymbol{\Sigma}_*^{(k)})$. Small · markers show the test set, and larger • markers show the training set.

Table 2
Model performance metrics when $K = 4$, including 80% curtailed data.

	Conventional regression		Mixture of regressions	
	RVM	GP	OMGP	Het-OMGP
NMSE	10.46	10.19	0.15	0.15
MSD	0.98	0.98	0.70	0.66

7.1. Validation: more turbines and curtailments

To demonstrate the flexibility of the model it is used to infer $K > 3$ latent functions, associated with a *separate* group of turbines in the wind farm. As before, the turbines exhibit normal, 50% curtailment, and zero-power relationships; however, an 80% curtailment is also observed. The priors of the OMGP are defined as in Section 6, with an additional soft-clip mean function component, such that $K = 4$. The number of components is verified via cross-validation in Appendix D. A total of 9973 observations are sampled from the data, corresponding to a (selected) subset of four turbines over seven weeks. Approximately $1/3$ of the data are used for training and $2/3$ for testing. A representative model is learnt for the alternative latent functions, visualised in Fig. 8. The same metrics are presented in Table 2 to highlight improvements. Again, the hyperparameters of the OMGP are interpretable: in particular, for the new curtailment $\alpha_1^{(k)} = 0.81$, corresponding approximately to 80%.

The validation experiments with four components ($K = 4$) highlight that the OMGP can be used to represent a variety of curtailment relationships, supporting modelling and monitoring procedures for a wide range of data that should be expected in practice.

7.2. Towards population-based monitoring: entropy measures

Considering applications of monitoring *in situ*, the OMGP can be used to inform novelty detection and classification across the wind farm. Novel observations of wind speed and power (from the full 125 week monitoring period) can be compared to the OMGP. This approach to performance monitoring is an approach in population-based SHM, whereby a general model, referred to as the *form* [24], is used to represent the behaviour of members within a population. In this case, the form is the OMGP and the population is the wind farm.

When monitoring via the power curve, the error given the predicted output (e.g. Eqs. (48) and (49)) can be used for novelty detection, as in [2,14,24]. Alternatively, with the OMGP, given wind speed and power observations, measurements can be classified using $\hat{k}_*$(43). Additionally, the distribution $p(k_* \mid \mathbf{x}_*, \mathbf{y}_*, \mathcal{D}) = P(k_* \mid \mathbf{x}_*, \mathbf{y}_*, \mathcal{D})$ (41) is informative from a monitoring perspective; this is the probability that $\{\mathbf{x}_*, \mathbf{y}_*\}$ were generated by component $f^{(k)}$ in the mixture. In other words, the probability that new data correspond to:

- the normal curve ($k_* = 1$),
- 50% curtailment ($k_* = 2$),
- or zero-power ($k_* = 3$).

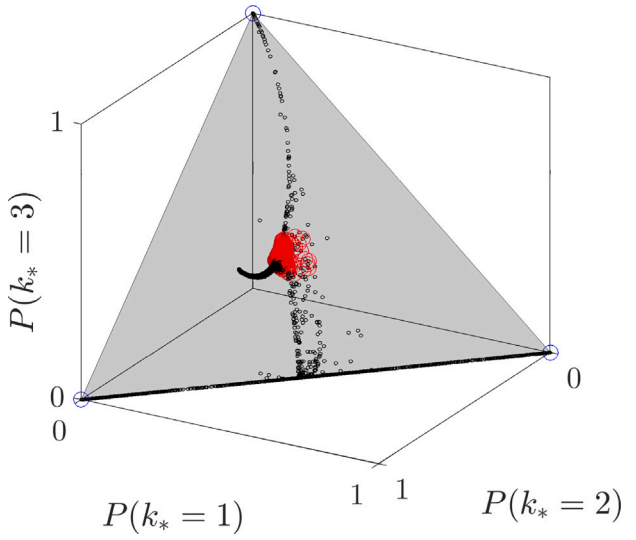

Fig. 9. The simplex (grey triangle) associated with the distribution $P(k_* \mid \mathbf{x}_*, \mathbf{y}_*, \mathcal{D})$. Points on the simplex represents observations of wind speed and power. Blue ∘ markers highlight low entropy points, red ∘ markers highlight high entropy points.

As $k_* \in \{1, 2, 3\}$, and $\sum_{k_*}^{K} P(k_* \mid \mathbf{x}_*, \mathbf{y}_*, \mathcal{D}) = 1 \quad \forall\{\mathbf{x}_*, \mathbf{y}_*\}$, it is possible to view power curve data as points on a 3D simplex, associated with the multinomial distribution $p(k_* \mid \mathbf{x}_*, \mathbf{y}_*, \mathcal{D})$. The grey triangle in Fig. 9 visualises the simplex where points are observations from the test set (concerning the 50% curtailment data). Although initially abstract, the plot is insightful from a monitoring perspective. It indicates that classes one and two (ideal and curtailed trends) are regularly confused, while class three (zero power) is equally confused with the others. This makes sense when inspecting Fig. 7: the ideal and curtailed trends are similar up to a normalised wind speed of zero, while the zero-power trend is indistinguishable from $k = 1$ and $k = 2$ at low wind speeds.

Given this distribution, the Shannon-entropy can be used as a measure of *uncertainty* to indicate if it is likely that new data were generated by latent functions within the OMGP,

$$H(k_*) = -\sum_{j=1}^{K} P(k_* = j \mid \mathbf{x}_*, \mathbf{y}_*, \mathcal{D}) \log P(k_* = j \mid \mathbf{x}_*, \mathbf{y}_*, \mathcal{D}) \tag{50}$$

With regard to the simplex in Fig. 9, each corner of the triangle relates to *low* entropy, corresponding to data that are classified with *certainty* (as $k_* = 1$, $k_* = 2$, or $k_* = 3$). On the other hand, the centre corresponds to *high* entropy, i.e. observations for which each component is equally likely (or none at all). During monitoring, high entropy data can be investigated, as it is unclear which component generated them. Examples of high and low entropy data given the test set are shown by red and blue markers respectively in Fig. 9. Following investigation, if it appears that new data correspond to an additional latent function (not yet included in the *form* of the wind farm) the mixture can be updated accordingly by adding a component, such that $K \leftarrow K+1$. Ideas behind modelling and updating the *form* for a wind farm population (and subsequent monitoring) are the focus of future work.

8. Conclusions

A novel data-driven model for wind turbine power data has been proposed. Critically, the method is capable of representing wind/power measurements including both *curtailed* and ideal operation. This is an alternative to the conventional approach, which filters out (and removes) the curtailed (SCADA) data. Consequently, the model should be representative of *in situ* behaviour, rather than ideal operation only.

A mixture of Gaussian processes infers *multivalued* wind-power relationships without labels to associate data to functions. Each function corresponds to a different operational condition (power curve) for a wind farm population. The algorithm is unsupervised, as labels to define which trend (ideal, curtailed, etc.) generated each of the measurements are not required; this information was not available in the experiments here. For each function in the mixture, input dependent noise is considered, a critical consideration when modelling power curve data. The model is applied to measurements from an operational wind farm, and it is shown to generalise well, representing future measurements from the population for various sets of turbines and curtailments. Finally, ideas for population-based power curve monitoring procedures (considering entropy measures) are introduced and discussed.

CRediT authorship contribution statement

L.A. Bull: Conceptualisation, Methodology, Software, Formal analysis, Writing – original draft. **P.A. Gardner:** Methodology, Writing – review & editing. **T.J. Rogers:** Methodology, Writing – review & editing. **N. Dervilis:** Methodology, Data curation, Writing

– review & editing, Supervision, Funding acquisition. **E.J. Cross:** Methodology, Writing – review & editing. **E. Papatheou:** Data curation, Methodology. **A.E. Maguire:** Data curation. **C. Campos:** Data curation. **K. Worden:** Conceptualisation, Data curation, Writing – review & editing, Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors gratefully acknowledge the support of the UK Engineering and Physical Sciences Research Council (EPSRC) through Grant references EP/R003645/1, EP/R004900/1, EP/R006768/1.

Appendix A. Type-II maximum likelihood

Gaussian process hyperparameters, $\boldsymbol{\theta} = \{\beta, \boldsymbol{\alpha}, \sigma_f, l, \sigma\}$, are (typically) optimised through empirical Bayes. This involves maximising the marginal likelihood of the model,

$$\begin{aligned} p(\mathbf{y} \mid \mathbf{x}\,;\,\boldsymbol{\theta}) &= \int p(\mathbf{y} \mid \mathbf{x}, \mathbf{f}) p(\mathbf{f} \mid \mathbf{x})\, d\mathbf{f} \\ &= \mathcal{N}\left(\mathbf{y}\,;\,\mathbf{m},\,\mathbf{K_{xx}} + \mathbf{R}\right) \end{aligned} \tag{A.1}$$

By marginalising (integrating) out the latent function values $\mathbf{f}$, this moves a level up the Bayesian hierarchy — mitigating issues of overtraining through the Bayesian Occam's razor [19]. An optimisation of this objective should lead to a minimally-complex model given the observed training data; for convenience and numerical stability, this is implemented as a minimisation of the negative-log-marginal-likelihood w.r.t $\boldsymbol{\theta}$,

$$\hat{\boldsymbol{\theta}} = \operatorname{argmin}_{\boldsymbol{\theta}} \{-\log p(\mathbf{y} \mid \mathbf{x}\,;\,\boldsymbol{\theta})\} \tag{A.2}$$

$$\begin{aligned} -\log p(\mathbf{y} \mid \mathbf{x}\,;\,\boldsymbol{\theta}) &\triangleq -\log \mathcal{N}\left(\mathbf{y} \mid \mathbf{m},\,\mathbf{K_{xx}} + \mathbf{R}\right) \\ &= \frac{1}{2}(\mathbf{y}-\mathbf{m})^{\top}\left(\mathbf{K_{xx}} + \mathbf{R}\right)^{-1}(\mathbf{y}-\mathbf{m}) \ldots \\ &\quad + \frac{1}{2}\log\left|\mathbf{K_{xx}} + \mathbf{R}\right| + \frac{N}{2}\log 2\pi \end{aligned} \tag{A.3}$$

The terms in Eq. (A.3) have an interpretable meaning: the first is a data *fit* (or error) term, the second is a model complexity term, and the last is a constant [27].

Appendix B. Noise-process optimisation

Summarised from [23]:

1. given $\mathcal{D}$, learn an initial homoscedastic GP: G_1 with hyperparameters θ
2. given G_1, estimate the empirical (log) input-dependent noise $\mathbf{g}'$ at the inputs $\mathbf{x}$ using Eq. (19)
3. given $\{\mathbf{g}', \mathbf{x}\}$, learn the noise process: G_2 with hyperparameters $\boldsymbol{\zeta}$
4. given $\mathcal{D}$, estimate the heteroscedastic GP G_3, using G_2 to define $r(x_i)$ for the noise kernel
5. if not converged, set $G_1 \leftarrow G_3$, and go to step 2.

Appendix C. RVM benchmark

The Relevance Vector Machine (RVM) follows the implementation of Tipping [38], with a radial-basis function kernel,

$$k(x_i, x_j) = \exp\left\{-\gamma(x_i - x_j)^2\right\}$$

The hyperparameter γ of the kernel is determined by 5-fold cross-validation [27]. In the first (50% curtailed) experiments, an optimal value was $\gamma = 2$, while in the second experiments (80% curtailed) $\gamma = 1.6$.

Appendix D. Example cross-validation

The corrected lower bound (36) can be monitored to indicate an appropriate number of components for the OMGP. This is shown in Fig. D.10 for the model in Section 7.1. The maximum value of the lower bound corresponds to a mixture with four components ($K = 4$) – in correspondence with the prior intuition.

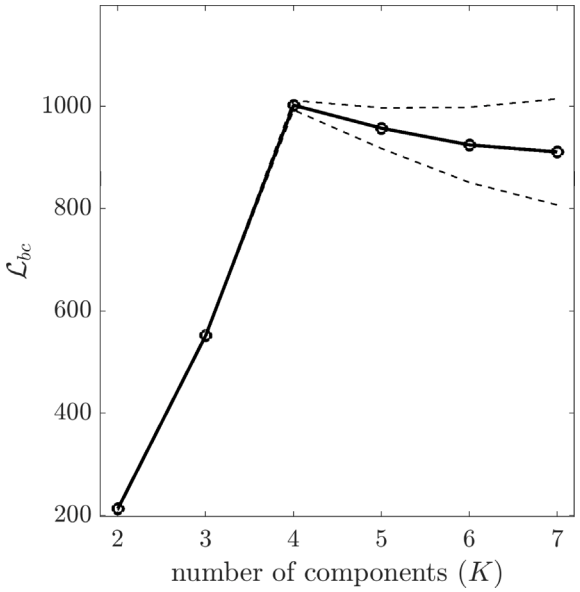

Fig. D.10. Monitoring the corrected lower bound ($\mathcal{L}_{bc}$) on the marginal likelihood while the number of components increases. The solid line represents the mean and the dashed line shows three-sigma standard deviation (12 repeats).

References

[1] 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.

[2] E. Papatheou, N. Dervilis, A. Maguire, C. Campos, I. Antoniadou, K. Worden, Performance monitoring of a wind turbine using extreme function theory, Renew. Energy 113 (2017) 1490–1502.

[3] W. Yang, R. Court, J. Jiang, Wind turbine condition monitoring by the approach of SCADA data analysis, Renew. Energy 53 (2013) 365–376.

[4] T. Rogers, P. Gardner, N. Dervilis, K. Worden, A. Maguire, E. Papatheou, E. Cross, Probabilistic modelling of wind turbine power curves with application of heteroscedastic Gaussian process regression, Renew. Energy 148 (2020) 1124–1136.

[5] V. Thapar, G. Agnihotri, V.K. Sethi, Critical analysis of methods for mathematical modelling of wind turbines, Renew. Energy 36 (11) (2011) 3166–3177.

[6] 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.

[7] 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.

[8] M. Waite, V. Modi, Modeling wind power curtailment with increased capacity in a regional electricity grid supplying a dense urban demand, Appl. Energy 183 (2016) 299–317.

[9] S.-h. Hur, W. Leithead, Curtailment of wind farm power output through flexible turbine operation using wind farm control, in: European Wind Energy Association Annual Event, EWEA 2014, 2014, pp. 1–9.

[10] M. Bontekoning, S.S. Perez-Moreno, B. Ummels, M. Zaaijer, Analysis of the reduced wake effect for available wind power calculation during curtailment, J. Phys. Conf. Ser. 854 (1) (2017) 012004.

[11] 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.

[12] A. Marvuglia, A. Messineo, Monitoring of wind farms' power curves using machine learning techniques, Appl. Energy 98 (2012) 574–583.

[13] 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.

[14] E. Papatheou, N. Dervilis, A. Maguire, I. Antoniadou, K. Worden, A performance monitoring approach for the Novel Lillgrund offshore wind farm, IEEE Trans. Ind. Electron. 62 (2015) 6636–6644.

[15] M. Lázaro-Gredilla, S.V. Vaerenbergh, N. Lawrence, Overlapping mixtures of Gaussian processes for the data association problem, Pattern Recognit. 45 (2012) 1386–1395.

[16] C. Rasmussen, C. Williams, Gaussian Processes for Machine Learning, The MIT Press, 2005.

[17] 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.

[18] T. Ouyang, A. Kusiak, Y. He, Modeling wind-turbine power curve: A data partitioning and mining approach, Renew. Energy 102 (2017) 1–8.

[19] C.E. Rasmussen, Z. Ghahramani, Occam's razor, in: Advances in Neural Information Processing Systems, 2001, pp. 294–300.

[20] X.-c. Fan, W.-q. Wang, R.-j. Shi, F.-t. Li, Analysis and countermeasures of wind power curtailment in China, Renew. Sustain. Energy Rev. 52 (2015) 1429–1436.

[21] G.-l. Luo, Y.-l. Li, W.-j. Tang, X. Wei, Wind curtailment of China's wind power operation: Evolution, causes and solutions, Renew. Sustain. Energy Rev. 53 (2016) 1190–1201.

[22] C.R. Farrar, K. Worden, Structural Health Monitoring: A Machine Learning Perspective, John Wiley & Sons, 2012.

[23] K. Kersting, C. Plagemann, P. Pfaff, W. Burgard, Most likely heteroscedastic Gaussian process regression, in: Proceedings of the 24th International Conference on Machine Learning, 2007, pp. 393–400.

[24] L. Bull, P. Gardner, J. Gosliga, T. Rogers, N. Dervilis, E. Cross, E. Papatheou, A. Maguire, C. Campos, K. Worden, Foundations of population-based SHM, Part I: Homogeneous populations and forms, Mech. Syst. Signal Process. 148 (2021) 107141.

[25] J. Gosliga, P. Gardner, L. Bull, N. Dervilis, K. Worden, Foundations of population-based SHM, Part II: Heterogeneous populations–Graphs, networks, and communities, Mech. Syst. Signal Process. 148 (2021) 107144.
[26] P. Gardner, L. Bull, J. Gosliga, N. Dervilis, K. Worden, Foundations of population-based SHM, Part III: Heterogeneous populations–Mapping and transfer, Mech. Syst. Signal Process. 149 (2021) 107142.
[27] K.P. Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2012.
[28] M.D. Klimek, M. Perelstein, Neural network-based approach to phase space integration, 2018, arXiv preprint arXiv:1810.11509.
[29] P.W. Goldberg, C.K. Williams, C.M. Bishop, Regression with input-dependent noise: A Gaussian process treatment, in: Advances in Neural Information Processing Systems, 1998, pp. 493–499.
[30] M. Lázaro-Gredilla, M.K. Titsias, Variational heteroscedastic Gaussian process regression, in: Proceedings of the 28th International Conference on International Conference on Machine Learning, 2011, pp. 841–848.
[31] D.M. Blei, A. Kucukelbir, J.D. McAuliffe, Variational inference: A review for statisticians, J. Amer. Statist. Assoc. 112 (518) (2017) 859–877.
[32] C. Tay, C. Laugier, Modelling smooth paths using Gaussian processes, in: Proceedings of the International Conference on Field and Service Robotics, Chamonix, France, 2007.
[33] J. Ross, J. Dy, Nonparametric mixture of Gaussian processes with constraints, in: International Conference on Machine Learning, PMLR, 2013, pp. 1346–1354.
[34] E. Snelson, Z. Ghahramani, Sparse Gaussian processes using pseudo-inputs, Adv. Neural Inf. Process. Syst. 18 (2005) 1257–1264.
[35] D.J. MacKay, D.J. Mac Kay, Information Theory, Inference and Learning Algorithms, Cambridge University Press, 2003.
[36] N.J. King, N.D. Lawrence, Fast variational inference for Gaussian process models through KL-correction, in: European Conference on Machine Learning, Springer, 2006, pp. 270–281.
[37] B. Jing, Z. Qian, A. Wang, T. Chen, F. Zhang, Wind turbine power curve modelling based on hybrid relevance vector machine, in: J. Phys. Conf. Ser., 1659 (1) (2020) 012034.
[38] M.E. Tipping, Sparse Bayesian learning and the relevance vector machine, J. Mach. Learn. Res. 1 (Jun) (2001) 211–244.