https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

Classification of Leading Edge Erosion Severity via Machine Learning Surrogate Models

Aidan Gettemy[1], Susan Minkoff[2], John Zweck[3], and Elaine Spiller[4]

[1]Department of Mathematical Sciences, University of Texas at Dallas, 800 W. Campbell Road, Richardson, TX 75080-3021
[2]Computing and Data Sciences Directorate, Brookhaven National Laboratory, 2 Center St, Upton, NY 11973
[3]Department of Mathematics, New York Institute of Technology, 1855 Broadway, New York, NY 10023
[4]Mathematical and Statistical Sciences, Marquette University, 1250 W. Wisconsin Ave., Milwaukee, WI 53233

Correspondence: Aidan Gettemy (Aidan.Gettemy@utdallas.edu)

Abstract.

As the number and size of wind turbines has increased, manual observation and maintenance of the turbines has become increasingly dangerous and time consuming for human operators. One key form of turbine deterioration is leading-edge erosion which degrades the blade laminate over time. This erosion is caused by environmental factors such as blowing sand, rain, and bug accumulation. Blade damage reduces aerodynamic efficiency and shortens the operational lifespan of wind turbines, motivating the need for structural health monitoring systems. Ideally one would like to use a digital twin which couples a physical device (the turbine) with a computer model by bidirectional passage of information between the physical and digital twins. In a digital twin, sensor data from the turbine continually updates the computer model which then predicts the state of the system for future maintenance and operation decisions, potentially eliminating the need for frequent manual inspections. Machine learning-based classifiers trained on simulation data accurately detect damage, but require large training data sets, highlighting the need for computationally efficient alternatives to full physics simulation. A Gaussian process (GP) surrogate model can be trained from a small set of full simulation datapoints. Once trained, GP's make predictions very fast (1000 times faster than a simulator evaluation) while also providing information about the uncertainty in the emulator prediction relative to the full physical simulator. The GP emulator methodology we employ includes two extensions to the standard GP. First, the output quantity of interest is vector-valued (rather than a scalar). In our case the vector contains statistics of relevant outputs such as lift, drag, generator power, etc. Second, the range of the outputs are constrained to fit specifications of the blade (so are not defined over the usual full-space domain required of Gaussian distributions). In this work we test two random forest classifiers developed to quantify levels of leading edge erosion. The classifiers differ in whether they are trained on full simulation data or data from the GP surrogate. We find that the classifier trained on surrogate data is as accurate as the classifier trained on full simulation data. Using the surrogate-generated dataset, the classifier distinguishes between five erosion severity levels with 87% accuracy, surpassing the simulation-trained classifier's accuracy of 83%. These results highlight the promise of using GP surrogates to train classifiers for leading edge erosion, a key component of a digital twin for wind turbine maintenance.

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

1 Introduction

Wind energy plays a crucial role in the wider international transition to renewable energy, with the European Union, China, and the United States planning to increase their supply of renewable power (Hassan et al., 2024). By 2050, wind energy could supply up to a third of global electricity demand (Bošnjaković et al., 2022). The increase in the number of offshore wind farms will contribute to lowering the cost of wind generated electricity by taking advantage of stronger, steadier winds over the ocean and utilizing larger rotor diameters (Bilgili and Alphan, 2022). However, the wind energy sector faces several challenges to delivering affordable, reliable power such as blade durability, large-scale turbine monitoring, and aging wind turbine infrastructure (Veers et al., 2019). Offshore turbines must endure a higher level of environmental damage than turbines on land, and both on and offshore turbines require higher levels of maintenance to address wear-related performance declines as they age. These factors contribute to operational expenses which represent a significant portion of power production costs (Park et al., 2022; Ren et al., 2021).

A major challenge in wind turbine maintenance is leading edge erosion (Maniaci et al., 2022). Rain, hail, and airborne particles roughen the blade's surface (Pugh et al., 2021), shifting the boundary between laminar and turbulent airflow (Panthi and Iungo, 2023) and compromising aerodynamic efficiency by reducing lift and increasing drag (Gaudern, 2014). Leading edge erosion forms as small pits merge over time into larger gouges, potentially leading to delamination (Sareen et al., 2014). This erosion of the laminate creates structural risks that may require the complete replacement of the blades of older turbines (Maniaci et al., 2022). Furthermore, newer rotors have faster erosion rates due to the increased blade tip wind speed inherent in longer blade designs (Carraro et al., 2022; Haus, 2020). Offshore turbines face an elevated risk for leading-edge erosion due to harsh environmental exposure (Shankar Verma et al., 2021b). Studies suggest erosion-related energy losses range from 2% to 3.7% annually (Han et al., 2018). These effects begin to accumulate within months of a blade's installation, with erosion accumulating faster at the tip (Pryor et al., 2022). Because erosion depends on coating fatigue and environmental conditions, predicting damage levels remains challenging (Pryor et al., 2025). Approaches to mitigate leading edge erosion include improving blade resistance (Antoniou et al., 2022; Herring et al., 2019), mapping environmental exposure to anticipate wear patterns (Macdonald et al., 2016; Pryor et al., 2022), and developing predictive models to estimate operation and maintenance costs (López et al., 2023; Shankar Verma et al., 2021a; Visbech et al., 2023). Since inspections require shutdowns and wind farms are increasingly difficult to access, especially in offshore conditions, continuous monitoring and damage detection are essential for erosion damage management that minimizes energy loss and avoids full blade replacement (Campobasso et al., 2023; Tchakoua et al., 2014).

Detecting leading-edge erosion remains challenging due to manufacturing defects (including defects in the blade coating) and environmental factors such as variation in precipitation intensity and wind speed (Mishnaevsky, 2022). Conventional detection methods such as visual inspection usually only detect defects after substantial degradation has occurred (Pugh et al., 2021). Newer detection methods including UAV-based blade surveillance (Shihavuddin et al., 2019) and gloss-based surface characterization (Leishman et al., 2022) may enhance the precision of erosion assessment but are not always simple to implement. Alternatively, Supervisory Control and Data Acquisition (SCADA) Data (Maldonado-Correa et al., 2020; Pandit et al.,

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

2023) is a readily available source of data, and new sensors, such as blade load measurements (Abdallah et al., 2015) and aerodynamic surface pressure (Duthé et al., 2021), allow for continuous erosion monitoring.

In wind energy, machine learning has been used to analyze SCADA data for turbine fault detection and blade damage assessment (Du et al., 2020; Stetco et al., 2019; Zaher et al., 2009). Machine learning has proven effective in enhancing damage identification (Jiménez et al., 2017). Various methods, including random forest (Fezai et al., 2020), XGBoost (Zhang et al., 2018), and neural networks (Chen et al., 2021; Choe et al., 2021), have successfully detected faults using condition monitoring data. Machine learning has also been applied to problems related to leading edge erosion detection. Campobasso et al. (2020) used a neural network to predict the loss of annual energy production from eroded blades. Computational fluid dynamics (CFD) coupled with aero-elastic modeling (OpenFAST) has been used to simulate erosion patterns and quantify damage using machine learning (Enríquez Zárate et al., 2022). Erosion has also been modeled as a stochastic process, employing transformer models to track erosion severity in turbulent conditions via aerodynamic pressure coefficients (Duthé et al., 2021). Deep learning methods have been applied to erosion prediction using aerodynamic data (Abdallah et al., 2022). Finally, machine learning models incorporating a variety of damage predictors including pressure, temperature, humidity, and wind speed have shown the advantage of using a variety of input channels for erosion detection in condition monitoring (Carmona-Troyo et al., 2025).

Machine learning presents a promising avenue for detecting leading-edge erosion, yet its effectiveness is constrained by the scarcity of high-fidelity training data (Ward et al., 2024). Operational turbine datasets are often noisy due to sensor faults and are not generally made publicly available by manufacturers and operators. To mitigate these limitations, numerical models are commonly used to generate synthetic training data for validating new condition monitoring approaches (Abdallah et al., 2022; Carmona-Troyo et al., 2025; Duthé et al., 2021; Enríquez Zárate et al., 2022). Since erosion is governed by stochastic environmental inputs, training robust detection algorithms from physically realistic simulations remains a computational bottleneck, limiting the scalability of condition monitoring systems. For instance, CFD captures the fine-scale aerodynamic effects of erosion but requires computationally expensive numerical solutions to the Navier–Stokes equations, rendering them infeasible at full turbine scales (Langel et al., 2017). Reduced-order approaches such as large eddy simulation and Reynolds-averaged Navier–Stokes alleviate some of the computational burden, but even simplified two-dimensional models remain prohibitively costly when simulating erosion under diverse atmospheric conditions (Carraro et al., 2022). Aero-elastic simulations offer a more tractable alternative, producing realistic SCADA and sensor data, but erosion classification remains sensitive to environmental variability. Studies demonstrate that the impact of blade damage is modulated by wind speed, turbulence intensity, and air density, necessitating extensive coverage of operational scenarios for reliable model generalization (Pandit et al., 2023; Papi et al., 2020). Generating comprehensive datasets with aero-elastic models for machine learning introduces additional computational bottlenecks (Duthé et al., 2021), particularly when these models are coupled with simulations for collision-induced damage from rain or hail (Fiore and Selig, 2016; Schramm et al., 2017).

Surrogate modeling addresses complexity and scalability challenges by providing a computationally efficient approach to damage tracking and predictive maintenance (Clark and Clark, 2022; Golparvar et al., 2021; Murcia et al., 2018; Singh et al., 2024). A promising direction for leading edge erosion detection involves integrating embedded sensors on blades with predic-

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

tive damage models and control mechanisms within a digital twin system (McGugan and Mishnaevsky Jr, 2020). Digital twins are virtual replicas of physical turbines which integrate real-time sensor data, physics-based models, and control mechanisms to enable predictive maintenance and automated decision-making (De Kooning et al., 2021; Fuller et al., 2020; Kapteyn et al., 2021). Potential applications include offshore wind turbine behavior modeling, global sensitivity assessments, and predictive maintenance planning. However, real time data integration and prediction require models which can be run extremely rapidly. The surrogate model is an integral component of the digital twin framework because it enables the accurate tracking of damage states that would otherwise be too costly to simulate.

In this paper we present a novel application of Gaussian process emulation for generating training data to classify leading edge erosion damage. Our Gaussian process-based surrogate emulates multiple quantities of interest with high predictive accuracy in significantly reduced time over full model simulation. In our experiments, the emulator results show (1) low normalized root mean squared error (<10%) and high credible interval coverage (>88%) on multi-modal sensor data; (2) improved classification accuracy for random forest classification when trained on emulated data compared to simulated data (87% vs. 83%); and (3) 1000+× faster predictions compared to aerodynamic simulations. While the accuracy improvement is modest, the computational savings are significant. Even accounting for running the simulator to produce the training data for the emulator, the overall time is drastically reduced as the GP does not require a large number of training points. By demonstrating that emulated data provides training quality comparable to full simulations, we establish the feasibility of surrogate models for leading edge erosion severity tracking. Efficient surrogates allow for the exploration of more complex turbine condition monitoring scenarios and new methods for maintenance optimization, valuable as the kernel of a digital twin for wind turbines.

2 Methods

In this section, we present a workflow for using emulation in wind turbine condition monitoring. We use OpenFAST (Golparvar et al., 2021; Jonkman et al., 2024; Moynihan et al., 2022; Murcia et al., 2018) for aero-elastic simulations. We further apply the Morris screening method (Morris, 1991) (a global sensitivity analysis tool) to identify key input parameters for surrogate model construction. For surrogate modeling, we use the Gaussian Process (GP) (Smith, 2024)) emulator to create an effective multi-modal surrogate for wind turbine simulations. We emphasize two extensions to the GP emulator that improve the emulator's fidelity and are novel in the context of wind energy modeling: parallel partial emulation (Gu and Berger, 2016) and the range-censored emulation (Spiller et al., 2023) (for range-limited outputs). We use a random forest classification algorithm (Breiman, 2001) for damage detection. Lastly we provide details on the construction of our experimental dataset which represents diverse environmental scenarios and erosion patterns for training and validation.

2.1 OpenFAST

The OpenFAST wind turbine simulator is developed and maintained by the National Renewable Energy Laboratory (NREL) for a wide range of aerodynamic and structural applications. OpenFAST simulates the response of a wind turbine to environmental

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

conditions by linking specialized structural and physical modules, including those for wind flow, aero-dynamics, structural dynamics, and servo-dynamics (Jonkman, 2013; Rinker et al., 2020).

OpenFAST links computational grids, aerodynamic forces, and control systems, which has facilitated its adoption in a broad set of onshore and offshore wind turbine reference model studies. Aslmostafa et al. (2024) couple the `AeroDyn` and `ServoDyn` modules to benchmark new controllers which maximize power for floating offshore wind turbines. Furthermore, OpenFAST can supply realistic aero-servo-elastic data for leading edge erosion detection, including the generation of blade-tip accelerometer data (Enríquez Zárate et al., 2022) and lift/drag sensor data (Duthé et al., 2021).

Because of its versatility and well-established capabilities, we use OpenFAST to generate realistic leading edge erosion data, relying on the coupling of four of the 14 main OpenFAST modules, `AeroDyn`, `ElastoDyn`, `ServoDyn`, and `InflowWind`. We selected these modules because they contribute to the simulation of blade forces including lift and drag, blade loads, generator power, and the wind environment. The `InflowWind` module calculates the wind vectors around the turbine while taking into account the turbine's size and hub height. `InflowWind` can simulate wind fields with a variety of properties. For example, uniform wind fields include wind shear (where the wind close to the ground has a different velocity than wind at the top of the wind turbine tower) and can simulate sudden changes in wind speed and wind direction. The simulation of aerodynamic forces and blade loads, handled by the module `Aerodyn`, is particularly important for leading edge erosion. `Aerodyn` uses blade element momentum theory (BEM) (Jonkman et al., 2015) to calculate aerodynamic forces on the rotor by breaking the blade into discrete regions or elements. These elements are assigned a lift or drag coefficient according to their instantaneous angle-of-attack with the oncoming air. The relationship between the angle-of-attack and lift or drag is modeled using the aerodynamic polar curve. OpenFAST uses a look-up table based on the polar curve to determine the lift and drag coefficient of each blade element as a function of time, enabling the realization of torque and thrust forces on the axial shaft of the wind turbine which then determine the generator power (Ning, 2014; López et al., 2023). `ServoDyn` regulates the electronic systems of the wind turbine using structural motions, loads, and wind measurements for inputs and calculates the power generation. It controls the actuators of the wind turbine model, which pitch the blades, regulate the rotor speed, and rotate the nacelle. In these simulations, the nacelle is fixed, but we enable the servo-controller to pitch the blades and regulate the rotor, in order to maximize power below rated wind speed and maintain rated power above rated wind speed. Aerodynamic changes due to leading edge erosion result in changes in the dynamics of the blade. These effects are modeled using the `ElastoDyn` module, which calculates the blade and tower displacement, velocity, and acceleration, and allows for the simulation of accelerometers at the tips of the blades and loads at the root of the blades. `ElastoDyn` uses several different input parameters, including the blade geometry, the mass or inertia of blade elements, and the stiffness of elements.

2.2 Gaussian process emulation

In this section we review Gaussian process (GP) emulation for surrogate modeling. For several decades GP emulation has been applied to problems from domains as varied as thermal energy storage (Currin, 1988), electrical circuits (Welch et al., 1992), and the design of chemical experiments (Sacks et al., 1989). Gaussian process emulators were introduced to reduce the computational cost of repeatedly evaluating complex numerical models. They are especially useful when the computational

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

cost of a single simulation is high and when an extremely large number of simulations are required. The GP emulator is trained on a limited number of simulations run at design points. Once trained, the GP acts as an interpolator between design points in input space, so that evaluating the output quantity of interest (QoI) at untested inputs is essentially free. Therefore, it is feasible to generate much larger training datasets for subsequent machine learning algorithms via an emulator than would be possible using the full simulator. The GP also provides uncertainty quantification for evaluations at new inputs via credible intervals which assess the accuracy of the emulator as a surrogate model for the full physical simulator.

For wind turbine condition monitoring, building training sets for damage classification requires evaluating the simulator hundreds to thousands of times, which can be computationally prohibitive. Steady state simulations in OpenFAST take on the order of minutes on a typical laptop/desktop computer, while turbulent wind and wave simulations in off-shore environments take much longer. GP's are good approximations of simulators that vary smoothly as a function of inputs, which is the case for wind turbines (Rinker, 2016). Further, prediction and uncertainty estimation are robust in the presence of limited data (as compared to neural networks, for example) (Myren and Lawrence, 2021).

2.2.1 Scalar Gaussian process emulation

We begin with a discussion of the basic GP formalism and then describe the extensions we employ for emulation of wind turbines. For scalar-valued output, the wind turbine simulator is treated as a function, f, from the space of input parameters, $\boldsymbol{u} \in \boldsymbol{\mathcal{U}} \subset \mathbb{R}^p$, to the output quantity of interest, $v \in \mathcal{V} \subset \mathbb{R}$. For wind turbine erosion modeling, inputs include (for example) the wind velocity and the blade damage level, with generator power as an output QoI. The unknown deterministic function $f : \boldsymbol{\mathcal{U}} \to \mathcal{V}$ is then modeled as a draw from a random Gaussian process (GP), $\tilde{f}$, with the property that for any finite subset, $\{\boldsymbol{u}_1, ..., \boldsymbol{u}_n\} \in \boldsymbol{\mathcal{U}}$, the random variables, $\{\tilde{f}(\boldsymbol{u}_1), ..., \tilde{f}(\boldsymbol{u}_n)\} \in \mathcal{V}$, follow a multivariate Gaussian distribution (Rasmussen, 2003). The modeling begins with the assumption of a GP prior with a mean trend $\mu : \boldsymbol{\mathcal{U}} \to \mathcal{V}$, and covariance, $C : \boldsymbol{\mathcal{U}} \times \boldsymbol{\mathcal{U}} \to \mathbb{R}$. The mean trend is of the form $\mu(\boldsymbol{u}) = \mathbf{h}^T(\boldsymbol{u})\boldsymbol{\theta}$ for some choice of q basis functions (often taken to be a constant or linear function of the input parameters), $\mathbf{h}(\boldsymbol{u}) = (h_1(\boldsymbol{u}), ..., h_q(\boldsymbol{u}))^T$, and regression coefficients, $\boldsymbol{\theta}$. The covariance is of the form $C(\boldsymbol{u}_1, \boldsymbol{u}_2) = \sigma^2 c(\boldsymbol{u}_1, \boldsymbol{u}_2)$, where σ^2 is the variance (output scaling) of the GP, and $c : \boldsymbol{\mathcal{U}} \times \boldsymbol{\mathcal{U}} \to \mathbb{R}$ is the correlation function, which is specified by the choice of a kernel (Rasmussen, 2003; Santner et al., 2003). In this work we use the Matérn $5/2$ kernel (Stein (1999), Gu et al. (2018), and Spiller et al. (2023)) given by $c(\boldsymbol{u}_1, \boldsymbol{u}_2) = \prod_{k=1}^{p}(1 + \sqrt{5}d_k + \frac{5}{3}d_k^2)\exp(-\sqrt{5}d_k)$. Here $d_k = \frac{1}{\gamma_k}|u_{1,k} - u_{2,k}|$ is a normalized distance between the k^{th} entries of the vectors $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$. The correlation coefficients, $\boldsymbol{\gamma} = (\gamma_1, \gamma_2, ..., \gamma_p)^T$, determine how rapidly the emulated output changes with respect to each input dimension. Letting $\boldsymbol{U}_i^{\mathrm{D}}$ $(i = 1, \ldots, n)$ represent the design of training inputs (e.g., different settings of wind speed and blade damage), the corresponding response vector of outputs (power generated for each setting) is given by $\boldsymbol{v}^{\mathrm{D}} = (v_1^{\mathrm{D}}, ..., v_n^{\mathrm{D}})^T$. We denote the training input/response data together as $\mathcal{D}_{\mathrm{GP}} = \{(\boldsymbol{U}_1^{\mathrm{D}}, v_1^{\mathrm{D}}), ..., (\boldsymbol{U}_n^{\mathrm{D}}, v_n^{\mathrm{D}})\}$.

The GP emulator's predictive posterior distribution evaluated at a new input, $v^* \sim \tilde{f}(\boldsymbol{u}^*)$, is conditioned on the training design/response, $\mathcal{D}_{\mathrm{GP}}$. It follows Student's t-distribution with $n - q$ degrees of freedom (Gu et al., 2018),

$$v^* \mid \boldsymbol{v}^{\mathrm{D}}, \boldsymbol{U}^{\mathrm{D}}, \boldsymbol{\gamma}, \boldsymbol{\theta}, \sigma^2 \sim \mathrm{St}\big(m(\boldsymbol{u}^*), \sigma^2 c''(\boldsymbol{u}^*), n - q\big). \tag{1}$$

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

The predictive mean, $m(\boldsymbol{u}^*)$, and predictive variance, $\sigma^2 c''(\boldsymbol{u}^*)$, are given by

$$m(\boldsymbol{u}^*) = \mathbf{h}^T(\boldsymbol{u}^*)\boldsymbol{\theta} + \mathbf{r}^T(\boldsymbol{u}^*)\boldsymbol{R}^{-1}(\boldsymbol{v}^{\mathrm{D}} - \mathbf{H}\boldsymbol{\theta}), \tag{2}$$

$$c''(\boldsymbol{u}^*) = 1 - \boldsymbol{r}^T(\boldsymbol{u}^*)\boldsymbol{R}^{-1}\boldsymbol{r}(\boldsymbol{u}^*) + \boldsymbol{w}(\boldsymbol{H}^T\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{w}^T, \tag{3}$$

where $\mathbf{R}$ is an $n \times n$ matrix of correlations between input design points, $\mathbf{R}_{i,j} = c(\boldsymbol{U}_i^{\mathrm{D}}, \boldsymbol{U}_j^{\mathrm{D}})$. Likewise $\mathbf{r}$ is a vector of correlations between the new input and each of the inputs in the design, $\mathbf{r}_i(\boldsymbol{u}^*) = c(\boldsymbol{u}^*, \boldsymbol{U}_i^{\mathrm{D}})$, and $\boldsymbol{w} = \left(\boldsymbol{r}^T(\boldsymbol{u}^*)\boldsymbol{R}^{-1}\boldsymbol{H} - \boldsymbol{h}^T(\boldsymbol{u}^*)\right)$.

To specify $\tilde{f}$, we need estimates of $\sigma^2, \boldsymbol{\theta}$, and $\boldsymbol{\gamma}$. Finding good estimates of the correlation lengths is key to fitting a GP emulator. We use the `R` package `RobustGASP` (Gu et al., 2019) which considers the marginal posterior density for $\boldsymbol{\gamma}$ to obtain the maximum-a-posteriori (MAP) estimate of $\boldsymbol{\gamma}$ (Gu et al., 2018). With $\boldsymbol{\gamma}$ in hand, we can estimate the trend parameters and the scalar variance respectively as,

$$\boldsymbol{\theta} = (\boldsymbol{H}^T\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{R}^{-1}\boldsymbol{v}^{\mathrm{D}}, \text{and} \tag{4}$$

$$\sigma^2 = (n-q)^{-1}(\boldsymbol{v}^{\mathrm{D}} - \boldsymbol{H}\boldsymbol{\theta})^T\boldsymbol{R}^{-1}(\boldsymbol{v}^{\mathrm{D}} - \boldsymbol{H}\boldsymbol{\theta}). \tag{5}$$

In the next two sections we introduce parallel partial emulation to handle vector-valued QoI's and range-limited emulation to handle QoI's with range constraints.

2.2.2 Parallel partial emulation

For simulators with vector-valued output, practitioners following the standard scalar GP methodology typically either train a separate GP for each component of the vector (Bayarri et al., 2015) or use dimension reduction on the output vector before fitting GPs (Higdon et al., 2008). The former approach can be computationally prohibitive, while the latter emulator is no longer an interpolator of the data. Instead, we use a parallel partial Gaussian process emulator (PPE) which approximates the entire output vector $f : \boldsymbol{\mathcal{U}} \to \mathbb{R}^s$ via a single emulator (Gu and Berger, 2016; Dolski et al., 2024). The PPE starts with the assumption that output components can be treated independently, each with their own scalar variance, σ_j^2, and trend parameters, $\boldsymbol{\theta}_j, j = 1, \ldots, s$. Yet all output components share a common correlation structure with respect to dependence on input scenarios.

For a design of n input scenarios, the vector-valued responses are now collected in an $n \times s$ matrix $\mathbf{V}^{\mathrm{D}}$ where the j^{th} column, $\mathbf{V}_j^{\mathrm{D}}$, corresponds to all n responses of the j^{th} output component in the training data. Much like the scalar GP, the PPE approximates the j^{th} component of $\boldsymbol{v}^* = \tilde{f}(\boldsymbol{u}^*)$ at an untested input $\boldsymbol{u}^*$ as a sample from the Student t-distribution with $n-q$ degrees of freedom given by

$$\boldsymbol{v}_j^* \mid \mathbf{V}_j^{\mathrm{D}}, \boldsymbol{U}^{\mathrm{D}}, \boldsymbol{\gamma}, \boldsymbol{\theta}_j, \sigma_j^2 \sim \mathrm{St}(m_j(\boldsymbol{u}^*), \sigma_j^2 c''(\boldsymbol{u}^*), n-q). \tag{6}$$

Here $\boldsymbol{\theta}_j$ is calculated by replacing $\boldsymbol{v}^{\mathrm{D}}$ by $\mathbf{V}_j^{\mathrm{D}}$ in Equation 4, and where $m_j(\boldsymbol{u}^*)$ and σ_j^2 are calculated by replacing $\boldsymbol{v}^{\mathrm{D}}$ by $\mathbf{V}_j^{\mathrm{D}}$ and $\boldsymbol{\theta}$ by $\boldsymbol{\theta}_j$ in Equations 2 and 5, respectively.

As each component of the output vector shares a common correlational structure, $\mathbf{R}$ is the only matrix which must be inverted. Thus, the cost of training and predicting with a PPE is comparable to that of a scalar GP.

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

2.2.3 Range limited emulation

Computer model QoIs are often required to be positive or to take values within a given interval, as is the case for wind turbine data. For example, for the NREL reference turbine, the generator power cannot exceed 5 MW, and the standard deviation of any random variable (sensor data) must remain positive. Restrictions on the range of f pose a challenge for GP emulation because Gaussian processes have full support. That is, a GP's predictive outputs take values in the interval from $(-\infty, \infty)$. Further, this severe form of non-stationarity (outputs varying smoothly as inputs change versus outputs not varying at all as inputs change), violates standard GP assumptions. To mitigate the difficulty of fitting a GP to a range-limited QoI, Spiller et al. (2023) proposed the *zero-censored* Gaussian process emulator (zGP).

Regardless of whether the scalar output v is in reality bounded above, $v \in (-\infty, v_{\max}]$, or bounded below, $v \in [v_{\min}, \infty)$, by applying a linear transformation, we can assume that the transformed output is bounded below by zero. The main objective of the zGP is to replace all of the zero outputs with negative values that are consistent with a GP fit only to the positive outputs. To start the process, zero-output training data is initialized with negative values (this can be done with a deterministic rule or by sampling GPs fit to positive data, see Algorithm 2 in Spiller et al. (2023) for more detail.) Then for each input in the design that led to a zero-output, a GP is fit to all other (imputed negative and positive) output responses. At the left-out design point, this GP's truncated Normal distribution (on $(-\infty, 0)$) is sampled. This sample replaces the previous negative sample for that design point in a Gibbs-sampled Markov Chain Monte Carlo (MCMC) scheme. The process is then repeated for each of the other design points that led to zero outputs to complete one step in the MCMC chain.

We observe that this imputation process (which can be expensive but is only done once as preprocessing) converges after a few thousand iterations. One hundred samples are kept from this chain (accounting for burn-in and thinning) for each input that originally led to zero output. For each such input, we take the average of the hundred negative MCMC samples as the final imputed negative response. We then use the imputed negative and original positive output response QoI's to fit a GP. For prediction at an untested input, we take the GP's prediction if positive, or zero if the GP's prediction is negative.

For vector-valued outputs, the zGP is a pre-processing step which is independently applied to each component of the output vector that is range constrained. This process can be sped up via parallelization. The resulting imputed values are used for the training of a PPE (see (Seidman, 2025)).

The parallel-partial zero-censored Gaussian process emulator (PPzGP) has several advantages that make it an effective surrogate model for wind turbine simulation. First, it predicts multiple outputs simultaneously, eliminating the need for multiple, independent models, and second, it satisfies known range constraints on sensor outputs. While scalar GP emulators have been used as surrogate models for analyzing wind turbine power production (Golparvar et al., 2021), maximizing power with wake steering (Gori et al., 2024), blade loads (Clark and Clark, 2022), and mono-pile reliability analysis (Morató et al., 2019), the results in this paper are the first where range-limited and parallel partial emulation have been combined for wind turbine modeling. The surrogate PPzGP model is orders of magnitude faster to interrogate than the full simulator, which means that it is tractable for generating large training data sets for damage classification.

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

2.3 Global sensitivity analysis

Global sensitivity analysis (GSA) identifies inputs which drive significant variation in the simulator outputs (Smith, 2024). By using GSA to fix non-influential inputs, fewer simulations are required to train an accurate emulator. Morris (1991) proposed the elementary effects method (Morris screening), as an efficient GSA method for high dimensional, non-linear and computationally expensive models. Velarde et al. (2019) used elementary effects analysis to determine the relative importance of input parameters for analyzing the foundation loads of offshore wind turbines.

To apply the elementary effects analysis the input space, $\boldsymbol{\mathcal{U}}$, is first scaled to the $p-$dimensional unit hypercube, which is partitioned into a grid with spacing Δ. A random sample of r initial input points, $\mathbf{u}_0^{(j)}$ for $j = 1, ..., r$, is taken from these grid points. Starting at each of these random points, a sequence of p points is generated by applying p unit changes to the input vectors in the entrants in a random order to create the elementary effects experiment design.

This design provides a more comprehensive sensitivity analysis than a one-factor-at-a-time (OAT) approach (Saltelli, 1999). Although it is less precise than variance-based global sensitivity analysis methods such as Sobol indices (Sobol, 2001), the Morris screening requires far fewer model evaluations. To explicitly quantify pairwise parameter interactions, methods like Sobol indices require a number of simulations which scales quadratically with the number of parameters, but the computational cost of the Morris screening scales linearly (Menberg et al., 2016).

For each component index l of $\mathbf{u}$ and each of the r trajectories indexed by j, the method calculates the elementary effect using a finite difference

$$EE_{l,j}(\mathbf{u}) = [f(\mathbf{u} + \mathbf{e}_l) - f(\mathbf{u})]/\Delta. \tag{7}$$

The mean and the standard deviation of the elementary effects, $\{EE_{l,1}, ..., EE_{l,r}\}$, are calculated for each dimension l. To avoid the possibility of large positive and negative elementary effects canceling out, we use $\mu_l^* = \frac{1}{r}\sum_{j=1}^{r} |EE_{l,j}|$, instead of the mean, μ_l. The standard deviation, which requires the average of elementary effects, μ_l, is given by $\sigma_l = \sqrt{\frac{1}{r}\sum_{j=1}^{r}(EE_{l,j} - \mu_l)^2}$ (Iwanaga et al., 2022). A large mean elementary effect, μ_l^*, indicates that the l^{th} input has a large overall effect, while a large σ_l indicates strong nonlinear effects or interactions with the other inputs (Cropp and Braddock, 2002). A plot of pairs (μ_l^*, σ_l) for each input $l \in \{1, ..., p\}$ is used to determine the relative importance of the inputs (Campolongo et al., 2007; Morris, 1991). We use the elementary effects tool provided in the Python Library `SALib` (Herman and Usher, 2017; Iwanaga et al., 2022).

2.4 Random forest classification algorithm

The full simulator and the PPzGP surrogate model map environmental and damage-related input parameters to an output vector that includes generator power, blade tip acceleration, and lift. Since erosion damage is challenging to directly observe on wind turbines in the field, the goal is to classify the damage level from the observed multi-modal sensor data. The level of damage due to leading edge erosion is stratified into several levels of erosion severity, $\{\alpha_1, .., .\alpha_k\}$. Random forests have been used for many problems ranging from biology (Boulesteix et al., 2012) and healthcare (Esmaily et al., 2018), to filtering sensor data (Buschjäger and Morik, 2017) and remote sensing (Belgiu and Drăguţ, 2016). Random forests have been applied to diagnose

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

wind turbine faults in real data (Fezai et al., 2020) and to detect irregular sensor patterns due to blade or gearbox damage (Zhang et al., 2018). Random forests handle mixed continuous and categorical input vectors, require low computational cost, provide internal error estimates on untested inputs, and even provide an ordering of the importance level of each of its inputs (Díaz-Uriarte and Alvarez de Andrés, 2006).

Random forests are built from individual decision trees. A decision tree for classification starts with the root node containing the full training dataset. It then bins data into smaller nodes (called leafs). Random forest classifiers use ensembles of decision trees which bin data into sets that have minimum variance. The predictions from the ensemble of decision trees are averaged for the final prediction. Points not used in building a particular decision tree (called out-of-bag samples) can be used to evaluate feature importance.

In this work, we use MatLab's random forest (`fitcensemble`) implementation and associated hyper-parameter optimization framework (`OptimizeHyperparameters`) to select the tree depth and the number of independent learners (The MathWorks Inc., 2024).

2.5 Leading edge erosion data generation

All datasets in this study were generated with the 5 MW reference wind turbine in OpenFAST using the `Aerodyn`, `Elastodyn`, `Servodyn`, and `Inflow Wind` modules (Jonkman et al., 2009). We assume constant wind speed, direction, air density, and shear (see Table 1). The default wind turbine controller model has variable speed and collective pitch control which regulates the power generated as a function of the wind speed by pitching the wind turbine blades and controlling the rotor torque to maximize power below the rated wind speed (Abbas et al., 2022).

Table 1. OpenFAST wind turbine simulation model parameters.

Simulation Parameter	Description
Power Rating	5 MW
Rotor Orientation, Configuration	Upwind, 3 Blades
Rotor, Hub Diameter	126 m, 3 m
Hub Height	90 m
Cut-in, Rated, Cut-out wind speed	$\{3, 11.4, 25\}$ ms^{-1}
Simulation Time Step	6.25 ms
Simulation Total Time	180 s
Wind Type	Uniform Wind Files

We varied four environmental inputs, wind direction, wind speed, air density, and wind shear. During each simulation, the turbine nacelle is fixed. The wind direction refers to the yaw angle measured from the nacelle. The wind speed ranges from the lowest speed the turbine generates power to the speed where the blades are stalled to prevent damage (cut-in to cut-out wind speeds) for the 5 MW reference wind turbine (see Table 1) (Jonkman et al., 2009). Because air density has a large impact on power (Golparvar et al., 2021), we use a wide range of air densities, based on a conservative lower bound expected in cold, low

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

Table 2. Environmental variable ranges

Input	Range	Units
Wind Direction	[-15, 15]	degrees
Air Density	[1.1, 1.42]	kgm^{-3}
Wind Speed	[3, 25]	ms^{-1}
Wind Shear Coefficient	[0, 0.5]	(-)

humidity conditions at around sea level up to realistic high temperatures for turbines in onshore environments (Liang et al., 2020). We model wind speed, V, as a function of height, h, above the ground using the wind shear power law (Platt et al., 2016) $V(h) = V_{\text{ref}} \left(\frac{h}{h_{\text{ref}}}\right)^{\nu}$ where V_r is the reference wind speed, h_{ref} is the hub height (see Table 1), and ν is the shear parameter. The nominal wind shear parameter depends on the topographic surroundings of the turbine. We set a value of $\nu = 0.2$ as the default wind shear parameter (Liu et al., 2024). In Table 2 we summarize the ranges of the environmental parameters we varied our inputs over for the design. To ensure that the turbine reaches equilibrium with the environmental conditions, we ran the simulations for 180 seconds and discarded the first two-thirds of simulation times to avoid transient effects (Golparvar et al., 2021).

2.5.1 Leading edge erosion model

We discretize the blade surface into six regions, defined by the data in Table 3 and shown in Figure 1. In the columns of Table 3 we show which `AeroDyn` blade nodes belong to each region, the location of the region along the blade, the length of the region, and the airfoil classes assigned to the region (which we kept the same as the NREL reference wind turbine (Jonkman et al., 2009)). We use a stochastic model for leading edge erosion in which the severity of damage falls into one of several classes (Maniaci et al. (2022)). Duthé et al. (2021) model erosion and Choe et al. (2021) model blade damage and structural damage by grouping the aerodynamic elements used by OpenFAST into different blade regions. Both Duthé et al. (2021) and Enríquez Zárate et al. (2022) capture aerodynamic changes due to leading edge erosion by altering the lift and drag coefficients of the airfoils in different regions along the blade.

In our leading edge erosion model, we group blades into five erosion severity classes defined by $\alpha = \{0.00, 0.25, 0.50, 0.75, 1.00\}$. The erosion level, $e_i \in [0, 1]$, of the i-th blade segment is a constant such that if $e_i = 0$, the blade element is "clean" and if $e_i = 1$ the blade element is at its maximal eroded state. Random erosion level vectors, $\mathbf{e}$, are sampled from a multivariate normal distribution whose mean vector and covariance matrix are given in terms of the erosion severity class, α,

$$\mathbf{e} \sim MVN(\boldsymbol{\mu}(\alpha), \boldsymbol{\Sigma}(\alpha)). \tag{8}$$

We approximate the aerodynamic effects of leading edge erosion on the blade by imposing a relationship between the blade region erosion level and the lift and drag coefficients. Han et al. (2018) found that the most severe case of erosion led to a lift loss of 53% while Sareen et al. (Sareen et al., 2014) saw an increase in drag of up to 500% for the most severely eroded airfoils. We, therefore, use the heuristic that the aerodynamic effects of erosion scale the lift coefficient in the i-th blade region

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

Table 3. Blade erosion regions, defined according to the aerodynamic nodes of the 5MW NREL reference turbine.

Region	Aerodyn Nodes	Distance from Hub	Region Length	Airfoils Included
1	5-6	10.25 m	8.2 m	Du40, Du35
2	7-8	18.45 m	12.3 m	Du35, Du30
3	9-10	30.75 m	8.2 m	Du25
4	11-12	38.95 m	8.2 m	Du21
5	13-14	47.15 m	7.5 m	NACA64
6	15-19 (18)	54.67 m	6.8 m	NACA64

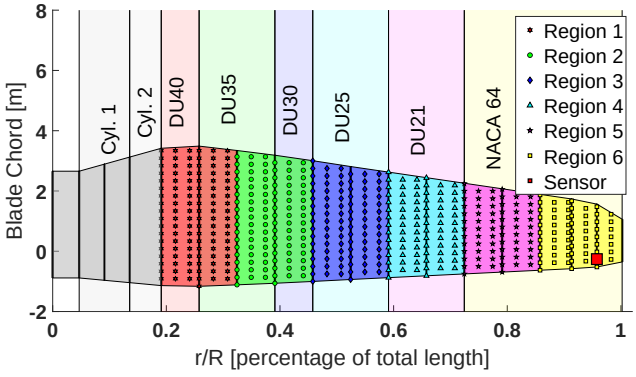

Figure 1. Diagram of erosion regions described in Table 4. The red square shows the location of the aerodynamic node where lift and drag sensor data is measured.

by $c_{L,i} = 1 - 0.53e_i$ and the drag coefficient by $c_{D,i} = 1 + 4e_i$. Since erosion accumulates faster and causes more damage at the tip of the blade (Keegan et al., 2013; Visbech et al., 2023), we set the mean erosion level vector to be $\boldsymbol{\mu}(\alpha) = (\alpha x_1, ..., \alpha x_6)$, where x_i is the distance from the hub to the midpoint of the i-th region, divided by the total blade length. The (i, j)-entry of the covariance matrix is

$$\boldsymbol{\Sigma}(\alpha)_{i,j} = \left(\frac{\min(x_i, x_j)\alpha}{10}\right)^2 \exp\left(\frac{-(x_i - x_j)^2}{x_i + x_j}\right). \tag{9}$$

This choice is justified by the observation that nearby regions have similar erosion levels, the variability of erosion accumulation increases towards the tip of the blade, and greater erosion severity is correlated with a longer exposure to erosion and hence a wider variability in erosion damage.

We use five outputs to detect erosion: generator power (Duthé et al., 2021; Enríquez Zárate et al., 2022), the lift and drag coefficient pressure sensors (Abdallah et al., 2022; Duthé et al., 2021), flap-wise acceleration of the tip of the blade (Du et al., 2020; Enríquez Zárate et al., 2022), and the edge-wise moment of the root of the blade (Moynihan et al., 2022; Sheibat-Othman et al., 2015). Note that we only track the sensor in the tip of the blade. Each output from our OpenFAST simulation is a vector with 160 entries per second of simulation time. Following a similar approach to Enríquez Zárate et al. (2022), we use statistical

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

Table 4. Wind turbine simulation outputs.

Output	OpenFAST	Unit	Description
Blade Tip Acceleration	TipALxB1	m s^{-2}	Blade local flapwise (absolute) tip acceleration.
Drag Sensor	B1N6Cd	(-)	Drag coefficient detected by Region 6 sensor.
Lift Sensor	B1N6Cl	(-)	Lift coefficient detected by Region 6 sensor.
Generator Power	GenPwr	MW	Generator Power.
Blade Root Moment	RootMxb1	kW-m	Blade Edgewise Moment (i.e., the moment caused by edgewise forces) at the blade root.

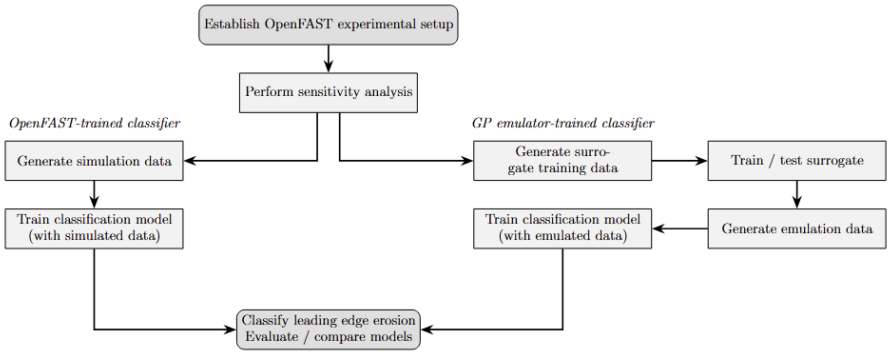

Figure 2. Workflow for developing the two leading edge erosion classifiers.

moments of each of the five output time-series, specifically the mean, standard deviation, skew, and kurtosis (Kaewniam et al., 2022), rather than the time-series itself.

In Table 4 we summarize the simulator outputs, along with their variable names in OpenFAST and their units. These quantities are the inputs to the random forest classifier which we use to determine erosion accumulation damage.

3 Workflow outline

Our goal in this work is to compare two random forest classifiers, one trained using OpenFAST-simulated data (500 points). The second trained using 5000 predictions from the PPzGP emulator. In Figure 2 we visualize the workflow for classifying leading edge erosion. After selecting an appropriate proxy for erosion, we identify influential input variables, and fix those that are non-influential. A primary dataset is simulated by OpenFAST and analyzed to determine the most important inputs for erosion damage classification. These inputs inform the quantities of interest that will be predicted by our PPzGP surrogate model. The two random forest classifers are compared to evaluate the effectiveness of the surrogate-based approach in Section 4.

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

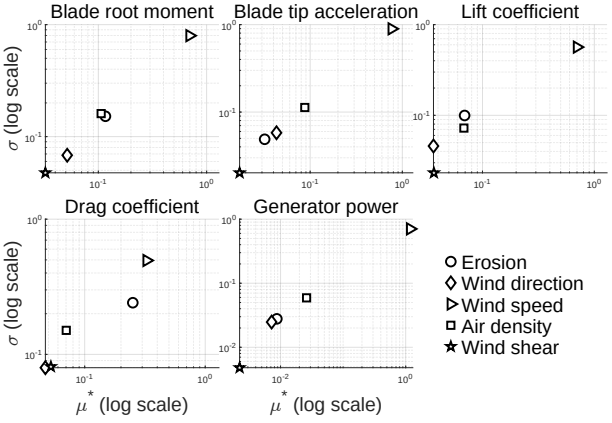

Figure 3. Elementary effects plots (logscale) for the five wind turbine QoI's.

4 Results

4.1 Global sensitivity analysis

The elementary effect analysis results show that wind shear has the least influence among the inputs tested, while wind speed has the most for each of the output QoI's. In Figure 3 we show the normalized elementary effect plots for each sensor output. For each graph, inputs with a large μ^* have a strong, linear effect on the output while those with a large σ interact nonlinearly with other inputs. We use a grid width of $\Delta = 1/9$ in the normalized input dimensions and $r = 25$ trajectories for a total of 150 simulations.

As wind shear has almost no effect on any of the output QoI's we considered, we fixed wind shear at the nominal value of 0.2. Since μ^* is far larger for wind speed than for air density, erosion, and wind direction, this sensitivity analysis supports the need for data from across a broad range of wind speeds for training erosion detection models (Papi et al., 2020). Importantly, the rankings of air density, erosion, and wind direction vary depending on the QoI. For instance, air density impacts generator power and root moment more than other non-wind-speed inputs. Air density is known to influence turbine generator power under the same constant wind speed conditions (Golparvar et al., 2021), as it affects the power available in the wind by increasing air mass interacting with the blades. (Specifically, $P = \frac{1}{2}\rho A v^3 C_p$, where ρ is air density, A is the swept area, v is wind velocity, and C_p is the power coefficient). Higher air density enhances lift force generation in blades, amplifying root bending moments and increasing power output. Wind direction, which is the relative angle of the wind to the nacelle direction, is the third most influential input for blade tip acceleration, but the fourth for every other output. This GSA study reveals that lift and drag are more sensitive to erosion inputs than any of the other outputs we tracked, supporting the use of aerodynamic pressure data for erosion detection (Abdallah et al., 2022; Carmona-Troyo et al., 2025; Duthé et al., 2021).

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

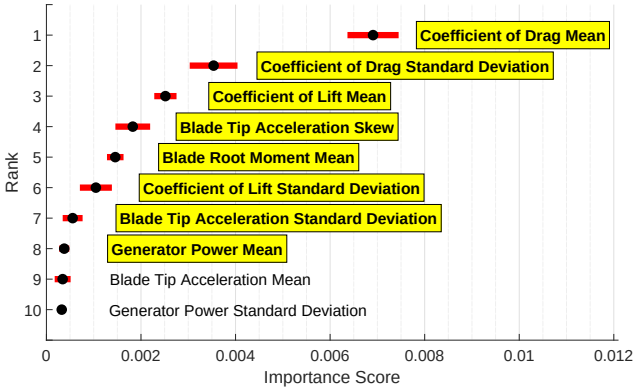

Figure 4. Mean importance score from random forest. Black dots represent the score over the 10 train/test splits. Corresponding standard deviation is given as red intervals.

4.2 Classifier input selection

Classification results often improve when the number of inputs is reduced. We use a separate initial random forest to perform this selection. To reduce bias, we repeat the predictor ranking calculation on 10 different train-test splits of the simulated training data. The training data for the random forest classification algorithm included 100 samples from each erosion class (drawn from a Latin hyper-cube design (Helton and Davis, 2003)). Wind speed, wind direction, air density along with the mean, standard deviation, skew, and kurtosis of the QoI's listed in Table 4 are the potential damage classification inputs.

In Figure 4 we show a ranking of the damage predictors. The variability of the score assigned to each predictor is small, which indicates that the rank of each damage predictor is stable. These damage predictors are the expected value and standard deviation of the aerodynamic drag sensor, the expected value and standard deviation of the aerodynamic lift sensor, the standard deviation and skew of the blade tip acceleration, the expected value of the blade root moment and of the generator power. We use these quantities as the QoI's we predict using the PPzGP surrogate.

The elementary effects study and the classifier input ranking support the use of pressure coefficient sensor data for erosion detection and underscore the importance of multi-modal data for accurate damage classification. The three most influential damage classification inputs were the mean aerodynamic drag coefficient sensor, the standard deviation of aerodynamic drag sensor, and the mean aerodynamic lift sensor.

4.3 Emulator evaluation

The PPzGP emulator is trained and evaluated on 210 total OpenFAST simulations (a separate data set from the one we use for classification). This dataset includes 50 samples for each of the 4 eroded blade classes and 10 for the clean blade class. The environmental inputs were selected using Latin hypercube sampling over the ranges in Table 2. The erosion level ranges are given in Table 5.

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

Table 5. Erosion level ranges for the six blade regions and five erosion severity classes in the training data for the PPzGP.

Severity Class ($\mathcal{A}$)	e_1	e_2	e_3	e_4	e_5	e_6
α_1 = 0.00	{0}	{0	{0}	{0}	{0}	{0}
α_2 = 0.25	[0.01, 0.02]	[0.04, 0.07]	[0.07, 0.11]	[0.12, 0.15]	[0.14, 0.19]	[0.18, 0.25]
α_3 = 0.50	[0.03, 0.04]	[0.09, 0.13]	[0.15, 0.22]	[0.21, 0.31]	[0.29, 0.39]	[0.36, 0.51]
α_4 = 0.75	[0.04, 0.07]	[0.13, 0.20]	[0.23, 0.32]	[0.32, 0.46]	[0.42, 0.59]	[0.55, 0.77]
α_5 = 1.00	[0.06 0.09]	[0.17 0.26]	[0.30 0.44]	[0.43 0.61]	[0.57 0.78]	[0.73 1.00]

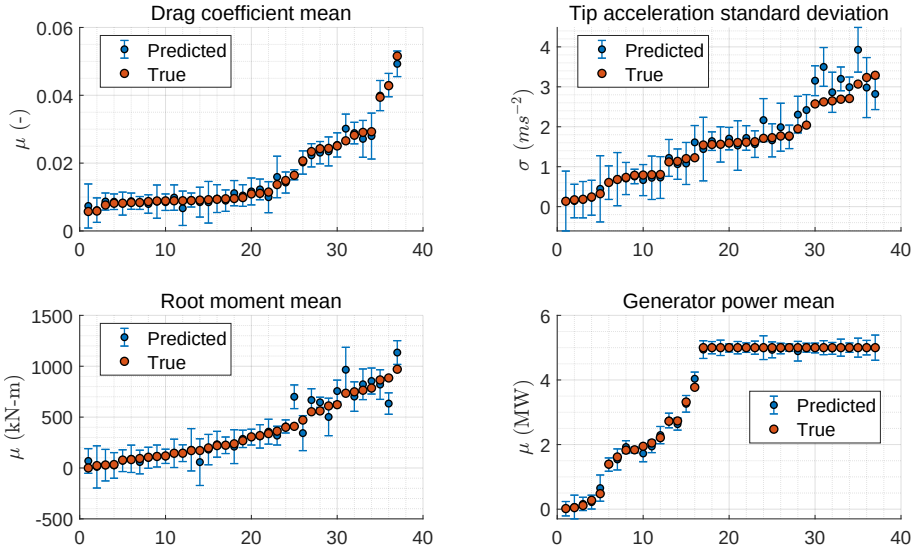

Figure 5. Predictions, ordered by simulated (true) value, and 95% credible intervals of the PPzGP surrogate model for four of eight outputs. Clockwise from top left: drag sensor mean, tip acceleration standard deviation, generator power mean, and root moment mean.

We test the accuracy of the PPzGP predictions using five distinct train/test splits of the simulated dataset. Each split allocates 173 training points proportionally across erosion severity classes, with the remaining 37 data points reserved for testing. In Figure 5, we compare the PPzGP emulator predictions (blue) with the simulated outputs (red), and we display the associated 95% credible intervals for four QoI's. For each output, the majority of true values fall within the credible intervals, indicating good surrogate model calibration. We note that the drag sensor mean and generator power predictions exhibit narrow credible intervals. Predictions for large root moments show wider intervals, reflecting increased uncertainty in this QoI. The flat plateau in the generator power predictions corresponds to the model respecting the rated power limit, while the physical constraint that the blade tip acceleration standard deviation is non-negative is similarly enforced. These model attributes are possible due to the zGP formulation and ensure that the PPzGP's outputs are physically meaningful.

In the left column of Table 6, we show the average percentage of predictions falling within the 95% credible intervals of the PPzGP emulator for each of the eight outputs across five training/test splits. While the expected coverage is around 95%

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

Table 6. Mean and standard deviation of the proportion of PPzGP predictions that fall within the 95% credible interval.

	Mean	Std
Coefficient of Drag mean	93.514	2.417
Coefficient Lift standard deviation	94.054	5.199
Coefficient of Lift mean	90.811	2.417
Tip Acceleration skew	87.568	8.674
Root Moment mean	80.000	6.784
Coefficient of Lift standard deviation	88.649	5.859
Tip Acceleration standard deviation	80.541	4.835
Generator Power mean	95.135	1.209

Table 7. Mean and standard deviation of the normalized root mean square errors of the PPzGP predictions.

	Mean	Std
Coefficient of Drag mean	0.075	0.028
Coefficient Lift standard deviation	0.103	0.025
Coefficient of Lift mean	0.029	0.007
Tip Acceleration skew	0.137	0.029
Root Moment mean	0.086	0.020
Coefficient of Lift standard deviation	0.079	0.031
Tip Acceleration standard deviation	0.098	0.032
Generator Power mean	0.025	0.004

based on the GP's estimated variance, observed coverage typically ranges between 80% and 95%. This discrepancy may be attributed to violations of the shared covariance assumptions inherent to the PPzGP model or to insufficient training data. In the right column, we show the standard deviation of the coverage, with the skew of the tip acceleration exhibiting the highest variability across splits. This is consistent with its elevated normalized root mean squared error (NRMSE). The results in the left column of Table 7, show that the average NRMSE is less than 10% for all but one output. The corresponding standard deviations shown in the right column are small, indicating consistent model performance across different data splits.

In Table 8, we show the computational time for the different stages of training and running the GP emulator. The pre-processing step for the range-limited outputs (see (Spiller et al., 2023)) is the most expensive part of building the PPzGP, but we can reduce this cost through parallelization. After this step, the actual training of the emulator takes a few seconds. Ultimately, taking predictions from the emulator is essentially free. A single simulation is on the order of 10 seconds, while one PPzGP evaluation is on the order of 0.001 of a second. (4 orders of magnitude faster than running the simulator).

In Figure 6 we show that the PPzGP surrogate captures the essential wind turbine dynamics represented by the full simulator. In the left panel, we show generator power as a function of wind speed across five erosion severity classes. The PPzGP

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

Table 8. PPzGP workflow computational time

Action	Time
Single simulation	46 s
PPzGP training simulations (173 points)	7900 s
PPzGP imputation (173 training points)	1400 s
PPzGp fitting (173 training points)	3.4 s
Single PPzGP prediction	0.0093 s

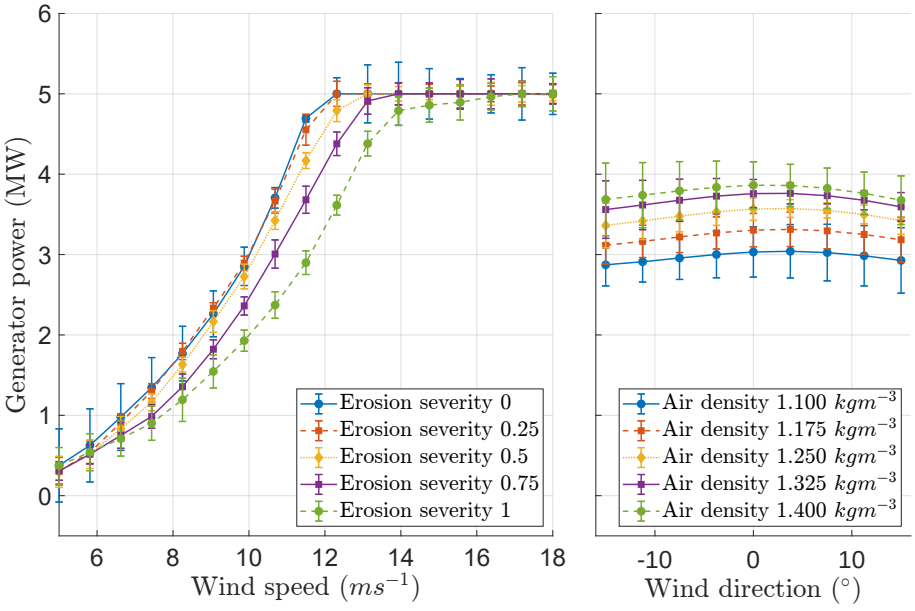

Figure 6. Left: Generator power as a function of wind speed for five levels of leading edge erosion. Right: Generator power as a function of wind direction for different air densities at fixed wind speed with clean blades.

accurately reproduces the power curve qualitatively, with higher erosion levels requiring greater wind speeds to achieve an equivalent power relative to the clean blades. The non-eroded power-curve aligns with findings from Golparvar et al. (2021), who used a GP to estimate the first and second statistical moments of generator power for a different turbine model. Unlike their single-output model, the PPzGP provides vector-valued predictions across multiple sensor outputs, with normalized RMSEs ranging from 2% to 14% (Table 7). In the right panel we demonstrate that the model captures two key physical trends: the power increases with air density and decreases when the rotor is misaligned with the wind (not at zero degrees). These relationships are consistent with prior studies (Hulsman et al., 2022) and are also evident in Figure 6. Power loss due to erosion is proportionally highest at lower power levels, remains significant up to the rated wind speed, and disappears beyond that threshold (Campobasso et al., 2023). However, as we see in Figure 5, model limitations exist. The prediction performance deteriorates

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

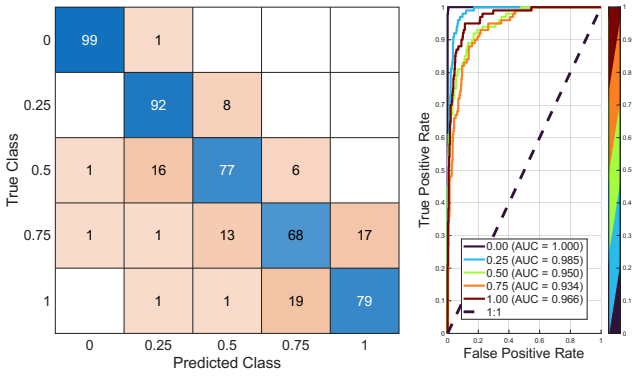

Figure 7. The left panel shows the confusion matrix for the classification model trained using simulated data and tested on simulated data. The right panel displays the ROC curves for each severity class.

in extreme loading conditions, particularly for high values of tip acceleration and root moment mean. These deficiencies likely stem from a lack of training data in more severe turbine operational states.

4.4 Classification comparison

The main results of our work are presented in this section in which we compare two random forest leading edge erosion classifiers. The first classifier is trained directly on simulated data from OpenFAST. The second is trained using data predicted by the PPzGP emulator. We use two simulated and one emulated dataset for these experiments. For the classifier trained on OpenFAST data, we use a design of 500 simulations for training and testing the classifiers. This dataset includes 100 samples from each of the five erosion severity classes where the wind speed, wind direction, and air density are varied within each class using a Latin hyper-cube design (see the ranges in Table 2). We train the emulator using a different set of 210 OpenFAST simulations (see the emulation training outlined in Section 2.5). The second random forest classifier is trained on emulator-predicted data which results from varying the environmental inputs in the same ranges, but uses 1000 data points per class instead of 100. Table 8 indicates that emulator predictions are very inexpensive compared to simulations. We compare the classification accuracy of the simulation-trained and emulation-trained damage classifiers using 10-fold cross-validation.

We quantify the accuracy of each classifier using the receiver operating characteristic (ROC) curve, which plots false positive rates on the horizontal axis and the true positive rates on the vertical axis. Larger values of the AUC metric, which is the area under the ROC curve, indicate higher accuracy of the classifier.

There are several conclusions we can draw from the simulation-trained results displayed in Figure 7. First, the simulation model is fairly accurate, predicting the correct label for between 68% and 100% of each class (diagonal boxes in blue). Most notably the class with the highest level of erosion is correctly predicted for 79 of the 100 samples. Second, the per-class AUC is high, above 0.93 for each class. The clean class has the highest AUC, while the intermediate damage classes (0.25, 0.50) are lower, which implies that the intermediate erosion severity levels are harder to distinguish. Third, incorrect classification is

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

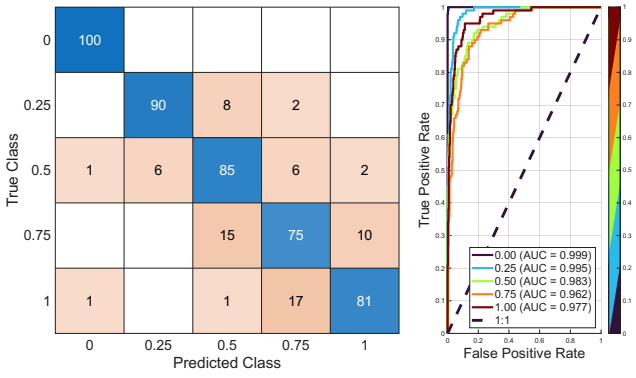

Figure 8. Corresponding results to those shown in Figure 7 for the classification model trained using emulated data.

Table 9. Comparison of the two classification models.

	Simulation-Trained	**Emulation-Trained**
AUC mean	0.9670	0.9834
AUC std	0.0265	0.0150
% Improvement	-	1.6970%
Accuracy mean	0.8300	0.8674
Accuracy std	0.0455	0.0669
% Improvement	-	4.5060%

most likely to occur between damage classes that are very close (off-diagonals on the confusion matrix). Indeed, only 5 out of 500 cases are misclassified by more than one damage level.

We compare the leading edge erosion classification results shown in Figure 7 to the emulation-trained random forest classifier results in Figure 8. We find that overall there is a slight increase in predictive accuracy observed for each class. Second, the AUC metrics are somewhat higher for the intermediate classes, where the simulation-trained classifier was the weakest. Overall, 7 out of 500 cases were misclassified by more than one damage level.We note that the most serious misclassification for either classifier occurs when the blade has an erosion level of 1 but is classified as being a clean blade (no erosion).

In Table 9, we summarize the results of the classification experiments. First, we highlight the AUC metrics (averaged across the 5 classes). The emulation-trained classifier has a slightly higher AUC on average. While the classification accuracy improvement is modest, the computational savings are significant. The PPzGP emulator (once trained) generates each random forest classifier training data sample in $\sim 0.01s$ as opposed to a single simulation run which takes $\sim 40s$. Even accounting for the 210 simulations we use to train the emulator, the overall computational effort of the complete PPzGP workflow is drastically reduced.

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

5 Discussion

5.1 PPzGP emulation

Surrogate modeling is gaining traction in wind turbine modeling. For instance, Branlard et al. (2020) used an augmented Kalman filter to predict tower-top loads with less than 10% error. Murcia et al. (Murcia et al., 2018) employed polynomial response surfaces to explore load uncertainty using only 140 simulations. Clark and Clark (2022) combined GPs with Bayesian Quadrature to estimate torque loading, optimizing training point placement to minimize posterior variance and save full-model interrogations. These examples reinforce the effectiveness of computationally efficient surrogate modeling.

The advantage of the PPzGP is its computational efficiency. The PPzGP accurately predicts multiple QoI's simultaneously while only requiring a relatively small training dataset. Our PPzGP emulator relies on the correlation between the different sensor outputs (they share the same input wind-field and structural coupling through the blades, hub, and tower) to produce multiple outputs using a single model, which eliminates the need to train and validate multiple independent surrogates. The surrogate also enforces physically meaningful bounds on output QoI's (such as generator power). Finally, we have demonstrated that our surrogate modeling approach can be used effectively for damage classification.

5.2 Damage classification

Structural health monitoring for leading edge erosion provides operators with the ability to make maintenance decisions that preserve integrity of the turbine (McGugan and Mishnaevsky Jr, 2020). The studies we present in this paper indicate that a random forest classifier is particularly effective at this prediction. Our random forest model trained on simulated data accurately classifies leading edge erosion in 83% of test cases. Classification accuracy is influenced by the number of erosion categories. Unsurprisingly, a model tasked with distinguishing between 4 classes instead of 10 showed a 30% increase in accuracy (Duthé et al., 2021). The machine learning model's ability to discriminate between the 5 damage classes we use allows for better decisions to be made regarding the operation and maintenance of the rotor and blades. Importantly, the emulator enables extremely efficient generation of more training samples which could lead to classifiers that are able to accurately distinguish between more erosion levels.

5.3 Damage predictors

Combining data from multiple sensors improves our ability to distinguish erosion effects from broader environmental influences. The use of additional sensor data is particularly valuable near rated wind speeds, where optimizing energy recovery is critical. For example, blade root moment (Moynihan et al., 2022; Sheibat-Othman et al., 2015) is a promising input for erosion classification. The selected classification features also include multiple statistical moments. Notably, both the mean and standard deviation of the aerodynamic drag coefficient give high importance scores (see Figure 4), suggesting that higher-order moments provide valuable information beyond mean values alone. This finding is consistent with the fact that wind approaching at an angle can activate structural modes, leading to periodic oscillations (Cardaun et al., 2019). In this study, periodic

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

outputs such as blade tip acceleration show that changes in standard deviation, and even skew, offer stronger predictive signals than the mean alone.

5.4 Future directions

There are several interesting future research directions for the PPzGP surrogate modeling framework and random forest classifiers. First, we did not consider expanding the PPzGP framework to emulate QoI's under turbulent wind conditions which occur naturally in the physical environment. While steady-state simulations already benefit from reduced computational cost, time-series simulations could lead to even greater efficiency gains.

The PPzGP emulator lays the foundation for a probabilistic, graph-based digital twin capable of tracking damage progression over time (Kapteyn et al., 2021). Digital twins inform control decisions that influence turbine operation such as adjusting tip speed during storms to reduce erosion (Bak et al., 2020; Pryor et al., 2025). While simulation demands were manageable in this study, complex applications, such as digital twin development, would benefit greatly from the speed-up enabled by surrogate modeling.

6 Conclusions

This study compares leading edge erosion classification using both a full physical simulator-trained random forest classifier and one trained by sampling a computationally efficient PPzGP emulator. We show that these two damage classifiers are equally accurate at predicting levels of leading edge erosion damage on wind turbine blades. The main contribution of this work is the use of the PPzGP emulator as a computationally efficient surrogate model for enabling damage classification tasks. The PPzGP can be trained on a relatively small set of full physical simulations. It can be adapted to handle arbitrarily large numbers of output QoI's without an increase in overall training cost. And the use of the range limited emulator (zGP) ensures that real-world turbine specifications can be incorporated into our model. Once trained, the emulator is several orders of magnitude faster to evaluate than the full simulator. Taken together, the advantages of the PPzGP framework lead to a reduction in computational demands and support real-time decision-making, such as that required for turbine digital twins. This study establishes the PPzGP-trained classifier as a promising tool with the potential to enhance operational decision-making and optimize turbine performance in dynamic environmental conditions.

Code and data availability. Code used to run the experiments and the data used for model training and testing are available on GitHub and at Zenodo https://doi.org/10.5281/zenodo.16729170 Gettemy (2025).

Author contributions. Conceptualization: AG, SM, and JZ. Simulation studies and investigation: AG. Methodology: AG, SM, and JZ. Writing (original draft preparation): AG, SM, JZ. Writing (review and editing): AG, SM, JZ, ES.

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

Competing interests. The authors declare they have no competing interests.

Acknowledgments

We would like to thank Todd Griffith for the initial suggestion for the project and guidance on wind turbine engineering. We are also grateful to Ipsita Mishra for her help learning how to run OpenFAST as well as useful discussions on wind turbine modeling in general. Aidan Gettemy was supported on this project by National Science Foundation Grant #2401945.

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

eawe european academy of wind energy
Open Access WIND ENERGY SCIENCE DISCUSSIONS

References

Abbas, N. J., Zalkind, D. S., Pao, L., and Wright, A.: A reference open-source controller for fixed and floating offshore wind turbines, Wind Energy Science, 7, 53–73, https://doi.org/https://doi.org/10.5194/wes-7-53-2022, 2022.

Abdallah, I., Natarajan, A., and Sørensen, J. D.: Impact of uncertainty in airfoil characteristics on wind turbine extreme loads, Renew. Energ., 75, 283–300, https://doi.org/https://doi.org/10.1016/j.renene.2014.10.009, 2015.

Abdallah, I., Duthé, G., Barber, S., and Chatzi, E.: Identifying evolving leading edge erosion by tracking clusters of lift coefficients, J. Phys. Conf. Ser., 2265, 032 089, https://doi.org/10.1088/1742-6596/2265/3/032089, 2022.

Antoniou, A., Dyer, K., Finnegan, W., Herring, R., Holst, B., Bech, J. I., Katsivalis, I., Kutlualp, T., Teuwen, J. J., et al.: Multilayer leading edge protection systems of wind turbine blades: a review of material technology and damage modelling, in: 20th European Conference on Composite Materials: Composites Meet Sustainability, EPFL Lausanne, Composite Construction Laboratory, Lausanne, Switzerland, https://doi.org/10.5075/epfl-298799_978-2-9701614-0-0, 26-30 June, 2022, 97–104, 2022.

Aslmostafa, E., Hamida, M. A., and Plestan, F.: Nonlinear control strategies for a floating wind turbine with PMSG in Region 2: A comparative study based on the OpenFAST platform, Ocean Eng., 300, 117 507, https://doi.org/https://doi.org/10.1016/j.oceaneng.2024.117507, 2024.

Bak, C., Forsting, A. M., and Sorensen, N. N.: The influence of leading edge roughness, rotor control and wind climate on the loss in energy production, J. Phys. Conf. Ser., 1618, 052 050, https://doi.org/doi:10.1088/1742-6596/1618/5/052050, 2020.

Bayarri, M. J., Berger, J. O., Calder, E. S., Patra, A. K., Pitman, E. B., Spiller, E. T., and Wolpert, R. L.: A Methodology for Quantifying Volcanic Hazards, International Journal of Uncertainty Quantification, 5, 297–325, https://doi.org/10.1615/Int.J.UncertaintyQuantification.2015011451, 2015.

Belgiu, M. and Drăguţ, L.: Random forest in remote sensing: A review of applications and future directions, ISPRS J. Photogramm., 114, 24–31, https://doi.org/https://doi.org/10.1016/j.isprsjprs.2016.01.011, 2016.

Bilgili, M. and Alphan, H.: Global growth in offshore wind turbine technology, Clean Technol. Envir., 24, 2215–2227, https://doi.org/https://doi.org/10.1007/s10098-022-02314-0, 2022.

Bošnjaković, M., Katinić, M., Santa, R., and Marić, D.: Wind turbine technology trends, Applied Sciences, 12, 8653, https://doi.org/https://doi.org/10.3390/app12178653, 2022.

Boulesteix, A.-L., Janitza, S., Kruppa, J., and König, I. R.: Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics, WIRES Data Min. Knowl., 2, 493–507, https://doi.org/https://doi.org/10.1002/widm.1072, 2012.

Branlard, E., Jonkman, J., Dana, S., and Doubrawa, P.: A digital twin based on OpenFAST linearizations for real-time load and fatigue estimation of land-based turbines, J. Phys. Conf. Ser., 1618, 022 030, https://doi.org/doi:10.1088/1742-6596/1618/2/022030, 2020.

Breiman, L.: Random forests, Machine learning, 45, 5–32, https://doi.org/https://doi.org/10.1023/A:1010933404324, 2001.

Buschjäger, S. and Morik, K.: Decision tree and random forest implementations for fast filtering of sensor data, IEEE T Circuits-I, 65, 209–222, https://doi.org/10.1109/TCSI.2017.2710627, 2017.

Campobasso, M. S., Cavazzini, A., and Minisci, E.: Rapid estimate of wind turbine energy loss due to blade leading edge delamination using artificial neural networks, J. Turbomach., 142, 071 002, https://doi.org/https://doi.org/10.1115/1.4047186, 2020.

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

Campobasso, M. S., Castorrini, A., Ortolani, A., and Minisci, E.: Probabilistic analysis of wind turbine performance degradation due to blade erosion accounting for uncertainty of damage geometry, Renew. Sust. Energ. Rev., 178, 113 254, https://doi.org/https://doi.org/10.1016/j.rser.2023.113254, 2023.

Campolongo, F., Cariboni, J., and Saltelli, A.: An effective screening design for sensitivity analysis of large models, Environ. Modell. Soft., 22, 1509–1518, https://doi.org/https://doi.org/10.1016/j.envsoft.2006.10.004, 2007.

Cardaun, M., Roscher, B., Schelenz, R., and Jacobs, G.: Analysis of wind-turbine main bearing loads due to constant yaw misalignments over a 20 years timespan, Energies, 12, 1768, https://doi.org/https://doi.org/10.3390/en12091768, 2019.

Carmona-Troyo, J. A., Trujillo, L., Enríquez-Zárate, J., Hernandez, D. E., and Cárdenas-Florido, L. A.: Classification of Damage on Wind Turbine Blades Using Automatic Machine Learning and Pressure Coefficient, Expert Systems, 42, e70 024, https://doi.org/https://doi.org/10.1111/exsy.70024, 2025.

Carraro, M., De Vanna, F., Zweiri, F., Benini, E., Heidari, A., and Hadavinia, H.: CFD modeling of wind turbine blades with eroded leading edge, Fluids, 7, 302, https://doi.org/https://doi.org/10.3390/fluids7090302, 2022.

Chen, H., Liu, H., Chu, X., Liu, Q., and Xue, D.: Anomaly detection and critical SCADA parameters identification for wind turbines based on LSTM-AE neural network, Renew. Energ., 172, 829–840, https://doi.org/https://doi.org/10.1016/j.renene.2021.03.078, 2021.

Choe, D.-E., Kim, H.-C., and Kim, M.-H.: Sequence-based modeling of deep learning with LSTM and GRU networks for structural damage detection of floating offshore wind turbine blades, Renew. Energ., 174, 218–235, https://doi.org/https://doi.org/10.1016/j.renene.2021.04.025, 2021.

Clark, A. C. and Clark, C. E.: Employing Bayesian Quadrature to Improve Fitting of Surrogate Models to Wind Turbine Loads, J. Phys. Conf. Ser., 2265, 042 045, https://doi.org/doi:10.1088/1742-6596/2265/4/042045, 2022.

Cropp, R. A. and Braddock, R. D.: The new Morris method: an efficient second-order screening method, Reliab. Eng. Syst. Safe., 78, 77–83, https://doi.org/https://doi.org/10.1016/S0951-8320(02)00109-6, 2002.

Currin, C.: A Bayesian approach to the design and analysis of computer experiments, Tech. rep., Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States), https://doi.org/https://doi.org/10.2172/814584, 1988.

De Kooning, J. D., Stockman, K., De Maeyer, J., Jarquin-Laguna, A., and Vandevelde, L.: Digital twins for wind energy conversion systems: a literature review of potential modelling techniques focused on model fidelity and computational load, Processes, 9, 2224, https://doi.org/https://doi.org/10.3390/pr9122224, 2021.

Díaz-Uriarte, R. and Alvarez de Andrés, S.: Gene selection and classification of microarray data using random forest, BMC bioinformatics, 7, 1–13, https://doi.org/https://doi.org/10.1186/1471-2105-7-3, 2006.

Dolski, T., Spiller, E. T., and Minkoff, S. E.: Gaussian process emulation for high-dimensional coupled systems, Technometrics, pp. 1–15, https://doi.org/https://doi.org/10.1080/00401706.2024.2322651, 2024.

Du, Y., Zhou, S., Jing, X., Peng, Y., Wu, H., and Kwok, N.: Damage detection techniques for wind turbine blades: a review, Mechanical Systems and Signal Processing, 141, 106 445, https://doi.org/https://doi.org/10.1016/j.ymssp.2019.106445, 2020.

Duthé, G., Abdallah, I., Barber, S., and Chatzi, E.: Modeling and monitoring erosion of the leading edge of wind turbine blades, Energies, 14, 7262, https://doi.org/https://doi.org/10.3390/en14217262, 2021.

Enríquez Zárate, J., Gómez López, M. d. l. Á., Carmona Troyo, J. A., and Trujillo, L.: Analysis and detection of erosion in wind turbine blades, Mathematical and Computational Applications, 27, 5, https://doi.org/https://doi.org/10.3390/mca27010005, 2022.

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

Esmaily, H., Tayefi, M., Doosti, H., Ghayour-Mobarhan, M., Nezami, H., and Amirabadizadeh, A.: A comparison between decision tree and random forest in determining the risk factors associated with type 2 diabetes, Journal of research in health sciences, 18, 412, https://doi.org/https://pmc.ncbi.nlm.nih.gov/articles/PMC7204421/, 2018.

Fezai, R., Dhibi, K., Mansouri, M., Trabelsi, M., Hajji, M., Bouzrara, K., Nounou, H., and Nounou, M.: Effective random forest-based fault detection and diagnosis for wind energy conversion systems, IEEE Sensors Journal, 21, 6914–6921, https://doi.org/10.1109/JSEN.2020.3037237, 2020.

Fiore, G. and Selig, M. S.: Simulation of damage progression on wind turbine blades subject to particle erosion, in: 54th AIAA Aerospace Sciences Meeting, San Diego, California, USA, https://doi.org/https://doi.org/10.2514/6.2016-0813, 4-6 January 2016, 0813, 2016.

Fuller, A., Fan, Z., Day, C., and Barlow, C.: Digital twin: enabling technologies, challenges and open research, IEEE access, 8, 108 952–108 971, https://doi.org/10.1109/ACCESS.2020.2998358, 2020.

Gaudern, N.: A practical study of the aerodynamic impact of wind turbine blade leading edge erosion, J. Phys. Conf. Ser., 524, 012 031, https://doi.org/10.1088/1742-6596/524/1/012031, 2014.

Gettemy, A.: Classification-of-Leading-Edge-Erosion-Severity-via-Machine-Learning-Surrogate-Models, https://doi.org/https://doi.org/10.5281/zenodo.16729170, 2025.

Golparvar, B., Papadopoulos, P., Ezzat, A. A., and Wang, R.-Q.: A surrogate-model-based approach for estimating the first and second-order moments of offshore wind power, Applied Energy, 299, 117 286, https://doi.org/https://doi.org/10.1016/j.apenergy.2021.117286, 2021.

Gori, F., Laizet, S., and Wynn, A.: Wind farm power maximisation via wake steering: a gaussian process-based yaw-dependent parameter tuning approach, Wind Energy, 27, 1545–1562, https://doi.org/https://doi.org/10.1002/we.2953, 2024.

Gu, M. and Berger, J. O.: Parallel partial Gaussian process emulation for computer models with massive output, Ann. Appl. Stat., pp. 1317–1347, https://doi.org/DOI: 10.1214/16-AOAS934, 2016.

Gu, M., Wang, X., and Berger, J. O.: Robust Gaussian stochastic process emulation, Ann. Stat., 46, 3038–3066, https://doi.org/https://doi.org/10.1214/17-AOS1648, 2018.

Gu, M., Palomo, J., and Berger, J. O.: RobustGaSP: Robust Gaussian Stochastic Process Emulation in R, The R Journal, 11, 112–136, https://doi.org/10.32614/RJ-2019-011, 2019.

Han, W., Kim, J., and Kim, B.: Effects of contamination and erosion at the leading edge of blade tip airfoils on the annual energy production of wind turbines, Renew. Energ., 115, 817–823, https://doi.org/https://doi.org/10.1016/j.renene.2017.09.002, 2018.

Hassan, Q., Viktor, P., Al-Musawi, T. J., Ali, B. M., Algburi, S., Alzoubi, H. M., Al-Jiboory, A. K., Sameen, A. Z., Salman, H. M., and Jaszczur, M.: The renewable energy role in the global energy transformations, Renew. Energ. Focus, 48, 100 545, https://doi.org/https://doi.org/10.1016/j.ref.2024.100545, 2024.

Haus, L. C.: New methods for digital twin modelling of wave and wind energy systems, Master's thesis, University of Texas at Dallas, https://doi.org/https://hdl.handle.net/10735.1/9016, 2020.

Helton, J. C. and Davis, F. J.: Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems, Reliab. Eng. Syst. Safe., 81, 23–69, https://doi.org/https://doi.org/10.1016/S0951-8320(03)00058-9, 2003.

Herman, J. and Usher, W.: SALib: An open-source python library for sensitivity analysis, The Journal of Open Source Software, 2, 97, https://doi.org/10.21105/joss.00097, 2017.

Herring, R., Dyer, K., Martin, F., and Ward, C.: The increasing importance of leading edge erosion and a review of existing protection solutions, Renew. Sust. Energ. Rev., 115, 109 382, https://doi.org/https://doi.org/10.1016/j.rser.2019.109382, 2019.

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

Higdon, D., Gattiker, J., Williams, B., and Rightley, M.: Computer model calibration using high-dimensional output, Journal of the American Statistical Association, 103, 570–583, https://doi.org/10.1198/016214507000000888, 2008.

Hulsman, P., Sucameli, C., Petrović, V., Rott, A., Gerds, A., and Kühn, M.: Turbine power loss during yaw-misaligned free field tests at different atmospheric conditions, J. Phys. Conf. Ser., 2265, 032 074, https://doi.org/10.1088/1742-6596/2265/3/032074, 2022.

Iwanaga, T., Usher, W., and Herman, J.: Toward SALib 2.0: advancing the accessibility and interpretability of global sensitivity analyses, Socio-Environmental Systems Modelling, 4, 18 155, https://doi.org/10.18174/sesmo.18155, 2022.

Jiménez, A. A., Muñoz, C. Q. G., and Márquez, F. P. G.: Machine learning for wind turbine blades maintenance management, Energies, 11, 1–16, https://doi.org/https://doi.org/10.3390/en11010013, 2017.

Jonkman, B., Platt, A., Mudafort, R. M., Branlard, E., Sprague, M., Ross, H., jjonkman, HaymanConsulting, Slaughter, D., Hall, M., Vijayakumar, G., Buhl, M., Russell9798, Bortolotti, P., reos rcrozier, Ananthan, S., RyanDavies19, S., M., Rood, J., rdamiani, nrmendoza, sinolonghai, pschuenemann, ashesh2512, kshaler, Housner, S., psakievich, Wang, L., Bendl, K., and Carmo, L.: OpenFAST/openfast: v3.5.3, https://doi.org/10.5281/zenodo.6324287, 2024.

Jonkman, J.: The new modularization framework for the FAST wind turbine CAE tool, in: 51st AIAA aerospace sciences meeting including the new horizons forum and aerospace exposition, Grapevine (Dallas/Ft. Worth Region), Texas, https://doi.org/https://doi.org/10.2514/6.2013-202, 7–10 January 2013, 202, 2013.

Jonkman, J., Butterfield, S., Musial, W., and Scott, G.: Definition of a 5-MW reference wind turbine for offshore system development, Tech. rep., National Renewable Energy Lab.(NREL), Golden, CO (United States), https://doi.org/https://doi.org/10.2172/947422, 2009.

Jonkman, J. M., Hayman, G., Jonkman, B., Damiani, R., Murray, R., et al.: AeroDyn v15 user's guide and theory manual, NREL Draft Report, 46, 2015.

Kaewniam, P., Cao, M., Alkayem, N. F., Li, D., and Manoach, E.: Recent advances in damage detection of wind turbine blades: a state-of-the-art review, Renew. Sust. Energ. Rev., 167, 112 723, https://doi.org/https://doi.org/10.1016/j.rser.2022.112723, 2022.

Kapteyn, M. G., Pretorius, J. V., and Willcox, K. E.: A probabilistic graphical model foundation for enabling predictive digital twins at scale, Nat. Computational Science, 1, 337–347, https://doi.org/https://doi.org/10.1038/s43588-021-00069-0, 2021.

Keegan, M. H., Nash, D., and Stack, M.: On erosion issues associated with the leading edge of wind turbine blades, J. Phys. D Appl. Phys., 46, 383 001, https://doi.org/10.1088/0022-3727/46/38/383001, 2013.

Langel, C. M., Chow, R. C., Van Dam, C., and Maniaci, D. C.: RANS based methodology for predicting the influence of leading edge erosion on airfoil performance, Tech. rep., Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), https://doi.org/https://doi.org/10.2172/1404827, 2017.

Leishman, G., Nash, D., Yang, L., and Dyer, K.: A novel approach for wind turbine blade erosion characterization: an investigation using surface gloss measurement, Coatings, 12, 928, https://doi.org/https://doi.org/10.3390/coatings12070928, 2022.

Liang, Y., Ji, X., Wu, C., He, J., and Qin, Z.: Estimation of the influences of air density on wind energy assessment: a case study from China, Energ. Convers. Manage., 224, 113 371, https://doi.org/https://doi.org/10.1016/j.enconman.2020.113371, 2020.

Liu, H., Chen, G., Hua, Z., Zhang, J., and Wang, Q.: Wind shear model considering atmospheric stability to improve accuracy of wind resource assessment, Processes, 12, 954, https://doi.org/https://doi.org/10.3390/pr12050954, 2024.

López, J. C., Kolios, A., Wang, L., and Chiachio, M.: A wind turbine blade leading edge rain erosion computational framework, Renew. Energ., 203, 131–141, https://doi.org/https://doi.org/10.1016/j.renene.2022.12.050, 2023.

Macdonald, H., Infield, D., Nash, D. H., and Stack, M. M.: Mapping hail meteorological observations for prediction of erosion in wind turbines, Wind Energy, 19, 777–784, https://doi.org/https://doi.org/10.1002/we.1854, 2016.

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

Maldonado-Correa, J., Martín-Martínez, S., Artigao, E., and Gómez-Lázaro, E.: Using SCADA data for wind turbine condition monitoring: a systematic literature review, Energies, 13, 3132, https://doi.org/https://doi.org/10.3390/en13123132, 2020.

Maniaci, D. C., MacDonald, H., Paquette, J., and Clarke, R. J.: Leading Edge Erosion Classification System, Tech. rep., Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), https://doi.org/https://doi.org/10.2172/2432094, 2022.

McGugan, M. and Mishnaevsky Jr, L.: Damage mechanism based approach to the structural health monitoring of wind turbine blades, Coatings, 10, 1223, https://doi.org/https://doi.org/10.3390/coatings10121223, 2020.

Menberg, K., Heo, Y., Augenbroe, G., and Choudhary, R.: New Extension Of Morris Method For Sensitivity Analysis Of Building Energy Models, https://publications.ibpsa.org/conference/paper/?id=bso2016_1101, 2016.

Mishnaevsky, L.: Root causes and mechanisms of failure of wind turbine blades: overview, Materials, 15, 2959, https://doi.org/https://doi.org/10.3390/ma15092959, 2022.

Morató, A., Sriramula, S., and Krishnan, N.: Kriging models for aero-elastic simulations and reliability analysis of offshore wind turbine support structures, Ships Offshore Struct., 14, 545–558, https://doi.org/https://doi.org/10.1080/17445302.2018.1522738, 2019.

Morris, M. D.: Factorial sampling plans for preliminary computational experiments, Technometrics, 33, 161–174, https://doi.org/https://doi.org/10.1080/00401706.1991.10484804, 1991.

Moynihan, B., Moaveni, B., Liberatore, S., and Hines, E.: Estimation of blade forces in wind turbines using blade root strain measurements with OpenFAST verification, Renew. Energ., 184, 662–676, https://doi.org/https://doi.org/10.1016/j.renene.2021.11.094, 2022.

Murcia, J. P., Réthoré, P.-E., Dimitrov, N., Natarajan, A., Sørensen, J. D., Graf, P., and Kim, T.: Uncertainty propagation through an aeroelastic wind turbine model using polynomial surrogates, Renew. Energ., 119, 910–922, https://doi.org/https://doi.org/10.1016/j.renene.2017.07.070, 2018.

Myren, S. and Lawrence, E.: A comparison of Gaussian processes and neural networks for computer model emulation and calibration, Stat. Anal. Data Min., 14, 606–623, https://doi.org/https://doi.org/10.1002/sam.11507, 2021.

Ning, S. A.: A simple solution method for the blade element momentum equations with guaranteed convergence, Wind Energy, 17, 1327–1345, https://doi.org/https://doi.org/10.1002/we.1636, 2014.

Pandit, R., Astolfi, D., Hong, J., Infield, D., and Santos, M.: SCADA data for wind turbine data-driven condition/performance monitoring: a review on state-of-art, challenges and future trends, Wind Engineering, 47, 422–441, https://doi.org/https://doi.org/10.1177/0309524X221124031, 2023.

Panthi, K. and Iungo, G. V.: Quantification of wind turbine energy loss due to leading-edge erosion through infrared-camera imaging, numerical simulations, and assessment against SCADA and meteorological data, Wind Energy, 26, 266–282, https://doi.org/https://doi.org/10.1002/we.2798, 2023.

Papi, F., Cappugi, L., Perez-Becker, S., and Bianchini, A.: Numerical modeling of the effects of leading-edge erosion and trailing-edge damage on wind turbine loads and performance, J. Eng. Gas Turb. Power, 142, 111 005, https://doi.org/https://doi.org/10.1115/1.4048451, 2020.

Park, J., Kim, C., Dinh, M.-C., and Park, M.: Design of a condition monitoring system for wind turbines, Energies, 15, 464, https://doi.org/https://doi.org/10.3390/en15020464, 2022.

Platt, A., Jonkman, B., and Jonkman, J.: InflowWind user's guide, Technical report, National Renewable Energy Laboratory, https://www.nrel.gov/docs/libraries/wind-docs/inflowwind_manual-pdf.pdf?sfvrsn=691bab12_1, 2016.

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

Pryor, S. C., Barthelmie, R. J., Cadence, J., Dellwik, E., Hasager, C. B., Kral, S. T., Reuder, J., Rodgers, M., and Veraart, M.: Atmospheric drivers of wind turbine blade leading edge erosion: review and recommendations for future research, Energies, 15, 8553, https://doi.org/https://doi.org/10.3390/en15228553, 2022.

Pryor, S. C., Coburn, J. J., and Barthelmie, R. J.: Spatiotemporal variability in wind turbine blade leading edge erosion, Energies, 18, 425, https://doi.org/https://doi.org/10.3390/en18020425, 2025.

Pugh, K., Nash, J., Reaburn, G., and Stack, M.: On analytical tools for assessing the raindrop erosion of wind turbine blades, Renew. Sust. Energ. Rev., 137, 110 611, https://doi.org/https://doi.org/10.1016/j.rser.2020.110611, 2021.

Rasmussen, C. E.: Gaussian processes in machine learning, in: Summer school on machine learning, pp. 63–71, Springer, https://doi.org/https://doi.org/10.1007/978-3-540-28650-9_4, 2003.

Ren, Z., Verma, A. S., Li, Y., Teuwen, J. J., and Jiang, Z.: Offshore wind turbine operations and maintenance: a state-of-the-art review, Renew. Sust. Energ. Rev., 144, 110 886, https://doi.org/https://doi.org/10.1016/j.rser.2021.110886, 2021.

Rinker, J., Gaertner, E., Zahle, F., Skrzypiński, W., Abbas, N., Bredmose, H., Barter, G., and Dykes, K.: Comparison of loads from HAWC2 and OpenFAST for the IEA wind 15 mw reference wind turbine, J. Phys. Conf. Ser., 1618, 052 052, https://doi.org/10.1088/1742-6596/1618/5/052052, 2020.

Rinker, J. M.: Calculating the sensitivity of wind turbine loads to wind inputs using response surfaces, J. Phys. Conf. Ser., 753, 032 057, https://doi.org/10.1088/1742-6596/753/3/032057, 2016.

Sacks, J., Schiller, S. B., and Welch, W. J.: Designs for computer experiments, Technometrics, 31, 41–47, https://doi.org/https://doi.org/10.1080/00401706.1989.10488474, 1989.

Saltelli, A.: Sensitivity analysis: Could better methods be used?, J. Geophys. Res.-Atmos., 104, 3789–3793, https://doi.org/https://doi.org/10.1029/1998JD100042, 1999.

Santner, T. J., Williams, B. J., Notz, W. I., and Williams, B. J.: The design and analysis of computer experiments, vol. 1 of *Springer Series in Statistics*, Springer, 2 edn., https://doi.org/https://doi.org/10.1007/978-1-4939-8847-1, 2003.

Sareen, A., Sapre, C. A., and Selig, M. S.: Effects of leading edge erosion on wind turbine blade performance, Wind energy, 17, 1531–1542, https://doi.org/https://doi.org/10.1002/we.1649, 2014.

Schramm, M., Rahimi, H., Stoevesandt, B., and Tangager, K.: The influence of eroded blades on wind turbine performance using numerical simulations, Energies, 10, 1420, https://doi.org/https://doi.org/10.3390/en10091420, 2017.

Seidman, J.: SideofMan/zGP: zGP in R v1.0.0, https://doi.org/10.5281/zenodo.17956672, 2025.

Shankar Verma, A., Jiang, Z., Ren, Z., Caboni, M., Verhoef, H., van der Mijle-Meijer, H., Castro, S. G., and Teuwen, J. J.: A probabilistic long-term framework for site-specific erosion analysis of wind turbine blades: A case study of 31 Dutch sites, Wind Energy, 24, 1315–1336, https://doi.org/https://doi.org/10.1002/we.2634, 2021a.

Shankar Verma, A., Jiang, Z., Ren, Z., Hu, W., and Teuwen, J. J.: Effects of onshore and offshore environmental parameters on the leading edge erosion of wind turbine blades: a comparative study, J. Offshore Mech. Arct., 143, 042 001, https://doi.org/https://doi.org/10.1115/1.4049248, 2021b.

Sheibat-Othman, N., Othman, S., Tayari, R., Sakly, A., Odgaard, P. F., and Larsen, L. F.: Estimation of the wind turbine yaw error by support vector machines, IFAC-PapersOnLine, 48, 339–344, https://doi.org/https://doi.org/10.1016/j.ifacol.2015.12.401, 2015.

Shihavuddin, A., Chen, X., Fedorov, V., Nymark Christensen, A., Andre Brogaard Riis, N., Branner, K., Bjorholm Dahl, A., and Reinhold Paulsen, R.: Wind turbine surface damage detection by deep learning aided drone inspection analysis, Energies, 12, 676, https://doi.org/https://doi.org/10.3390/en12040676, 2019.

https://doi.org/10.5194/wes-2025-289
Preprint. Discussion started: 28 January 2026

Singh, D., Dwight, R., and Viré, A.: Probabilistic surrogate modeling of damage equivalent loads on onshore and offshore wind turbines using mixture density networks, Wind Energy Science Discussions, 2024, 1–28, https://doi.org/https://doi.org/10.5194/wes-9-1885-2024, 2024.

Smith, R. C.: Uncertainty quantification: theory, implementation, and applications, SIAM, https://doi.org/https://doi.org/10.1137/1.9781611977844, 2024.

Sobol, I. M.: Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates, Math. Comput. Simulat., 55, 271–280, https://doi.org/https://doi.org/10.1016/S0378-4754(00)00270-6, 2001.

Spiller, E. T., Wolpert, R. L., Tierz, P., and Asher, T. G.: The Zero Problem: Gaussian Process Emulators for Range-Constrained Computer Models, SIAM/ASA Journal on Uncertainty Quantification, 11, 540–566, https://doi.org/10.1137/21M1467420, 2023.

Stein, M. L.: Interpolation of spatial data: some theory for kriging, Springer Science & Business Media, 1999.

Stetco, A., Dinmohammadi, F., Zhao, X., Robu, V., Flynn, D., Barnes, M., Keane, J., and Nenadic, G.: Machine learning methods for wind turbine condition monitoring: a review, Renew. Energ., 133, 620–635, https://doi.org/https://doi.org/10.1016/j.renene.2018.10.047, 2019.

Tchakoua, P., Wamkeue, R., Ouhrouche, M., Slaoui-Hasnaoui, F., Tameghe, T. A., and Ekemb, G.: Wind turbine condition monitoring: State-of-the-art review, new trends, and future challenges, Energies, 7, 2595–2630, https://doi.org/https://doi.org/10.3390/en7042595, 2014.

The MathWorks Inc.: Statistics and machine learning toolbox, https://www.mathworks.com/help/stats/index.html, 2024.

Veers, P., Dykes, K., Lantz, E., Barth, S., Bottasso, C. L., Carlson, O., Clifton, A., Green, J., Green, P., Holttinen, H., et al.: Grand challenges in the science of wind energy, Science, 366, eaau2027, https://doi.org/10.1126/science.aau2027, 2019.

Velarde, J., Kramhøft, C., and Sørensen, J. D.: Global sensitivity analysis of offshore wind turbine foundation fatigue loads, Renew. Energ., 140, 177–189, https://doi.org/https://doi.org/10.1016/j.renene.2019.03.055, 2019.

Visbech, J., Göçmen, T., Hasager, C. B., Shkalov, H., Handberg, M., and Nielsen, K. P.: Introducing a data-driven approach to predict site-specific leading-edge erosion from mesoscale weather simulations, Wind Energy Science, 8, 173–191, https://doi.org/https://doi.org/10.5194/wes-8-173-2023, 2023.

Ward, T., Jenab, K., Ortega-Moody, J., and Staub, S.: A comprehensive review of machine learning tchniques for condition-based maintenance, International Journal of Prognostics and Health Management, 15, https://doi.org/https://doi.org/10.36001/ijphm.2024.v15i2.3850, 2024.

Welch, W. J., Buck, R. J., Sacks, J., Wynn, H. P., Mitchell, T. J., and Morris, M. D.: Screening, predicting, and computer experiments, Technometrics, 34, 15–25, https://doi.org/https://doi.org/10.1080/00401706.1992.10485229, 1992.

Zaher, A., McArthur, S., Infield, D., and Patel, Y.: Online wind turbine fault detection through automated SCADA data analysis, Wind Energy, 12, 574–593, https://doi.org/https://doi.org/10.1002/we.319, 2009.

Zhang, D., Qian, L., Mao, B., Huang, C., Huang, B., and Si, Y.: A data-driven design for fault detection of wind turbines using random forests and XGboost, Ieee Access, 6, 21 020–21 031, https://doi.org/10.1109/ACCESS.2018.2818678, 2018.