Politecnico di Torino

Department of Mechanical and Aerospace Engineering

Master's degree in Mechanical Engineering

Data-Driven Diagnosis of Wind Turbines via SCADA-Based Anomaly Detection Methods

Supervisors:

Alessandro Fasana

Alessandro Paolo Daga

Luca Viale

Candidate:

Marco Gerbino

Academic year 2024/2025

Abstract

The increasing demand for wind energy requires reliable predictive maintenance strategies to minimise premature failures caused by mechanical phenomena such as wear. Therefore, the early detection of failures is essential to improve the efficiency of predictive maintenance and ensure continuous electrical energy production.

Over the last decade, data collected by the Supervisory Control and Data Acquisition (SCADA) system has become a common solution, especially due to its easily accessible data and cost-effective nature.

This thesis focuses on developing various machine learning methods to establish an anomaly detection strategy, starting with the computation of a regression model, and concluding in an anomaly index that enables the assessment of the turbine's health status. Specifically, this approach combines a physical analysis based on the Betz model with a machine learning technique, namely Support Vector Regression, to distinguish normal behaviour from potential faults.

Additionally, the advantages and main limitations of a univariate analysis are investigated, and a comparison with a multivariate analysis is also performed. The workflow performance is evaluated through the application of several performance metrics typically used in this field to assess the model's ability to detect anomalies under normal operating conditions. This standardised approach facilitates comparison with other techniques as defined in related research.

Finally, a classification method involving multiclass analysis is conducted, with the computation of an additional performance metric: class error.

An extract of this thesis was also presented during the SURVISHNO 2025 conference in Paris, under the title *Anomaly Detection in Wind Turbines under Operational Variability via SCADA and Residual Analysis*.

Contents

1. Introduction 13
1.1 Overview 13
1.2 Comparative Maintenance Strategies 16
1.3 Diagnostic Definition 17
1.4 Vibration Monitoring 20
1.5 Anomaly Detection 21
1.6 Machine Learning 24
1.6.1 Regression Methods 26
1.6.2 Dimensionality Reduction 28
1.6.3 Anomaly Detection in Machine Learning 29
1.7 Challenges and Recent Research 30
1.8 Thesis Objectives 32

2. Wind Turbines 34
2.1 Wind Turbines Overview 34
2.2 Main Components of Wind Turbines 36
2.3 Failure Mechanisms 38
2.3.1 Blade Failure 39
2.3.2 Gearbox Failure 41
2.3.3 Pitch System Failure 44
2.3.4 Yaw System Failure 44
2.4 Off-design Power Regulation Methods 46
2.5 Thesis Purposes 48

3. Dataset Description and Initial Preprocessing 50
3.1 Dataset 50
3.2 Data Acquisition 53
3.2.1 Generation of Dynamic Input 54
3.2.2 Sensor Instrumentation and Conditioning 55
3.2.3 Data Transmission and Digital Acquisition 57
3.2.4 Analog-to-Digital Conversion 57
3.3 Features Selection 58
3.4 Data Cleaning 63

4. Methodology 64
4.1 Method Overview 64
4.2 Betz Model 68
4.3 Linear Regression Models 70
4.4 Principal Component Analysis 74
4.5 Support Vector Regression 76
4.6 Regression Metrics Performance 79
4.7 Anomaly detection methods 81
4.8 Classification models adopted for multi-class analysis 84
4.9 Metrics Performance Anomaly Detection 87

5. Results 92
5.1 Statistical Analysis 92
5.2 Univariate Analysis 99
5.2.1 PCA Results 100
5.2.2 SVR Results: Training, Validation and Test 102
5.2.3 Performance Evaluation of Univariate-Based Anomaly Detection 109
5.3 Linear Regression Power-Wind Speed 115
5.4 Multivariate Analysis 119
5.5 Classification Results 126

6. Final Remarks and Future Work 130
Future Work 131

References 134

List of Figures

Figure 1.1: Percentage distribution of common failure modes in primary wind turbine components 14
Figure 1.2: General workflow adopted in a CBM program 17
Figure 1.3: Schematic representation of a damage detection process based on dynamic response comparison 19
Figure 1.4: Schematic representation of a SDOF system (left) and a MDOF system (right) 20
Figure 1.5: MIMO system 21
Figure 1.6: Point anomalies examples in a 2-dimensional space: O1, O2 and O3 are isolated observations that significantly deviate from the normal data distributions N1 and N2 23
Figure 1.7: Collective anomalies example in wind turbine's power output. In red is highlighted the anomaly 24
Figure 1.8: Machine learning taxonomy 25
Figure 1.9: Anomaly detection based on classification approach 30
Figure 2.1 :Horizontal axis wind turbines (left) and vertical axis wind turbine (right) 35
Figure 2.2: Horizontal axis wind turbine representation, and its main component groups 36
Figure 2.3: Planetary and two stages and gearbox configurations 37
Figure 2.4: Drivetrain representation 37
Figure 2.5: Main failure mechanisms in wind turbine 39
Figure 2.6: Schematic representation of boundary-layer flow over a region of surface roughness and its separation 40
Figure 2.7: Real images of wind turbine blades affected by different types of failure: structural fracture, lightning damage, and surface abrasion 40
Figure 2.8: Wear corrosion induced by chemical reactions in the lubricating oil 41
Figure 2.9: Micro-pitting on the gear tooth surface 42
Figure 2.10: Macro-pitting of tooth surface 42
Figure 2.11: Severe fretting corrosion on a gear tooth 43
Figure 2.12: Fractured gear tooth due to overload or fatigue 44
Figure 2.13: Two common failure modes in wind turbine yaw gears: (a) a broken tooth in the yaw ring gear; (b) multiple broken in the yaw drive gear 45
Figure 2.14: Solidity graph 46
Figure 3.1: Example images of offshore, onshore, and nearshore wind farms 51
Figure 3.2: Block diagram of the overall data acquisition workflow 53

Figure 3.3: Images of cup anemometers, propeller anemometers, and sonic anemometers 55
Figure 4.1: General Workflow including preprocessing Steps 64
Figure 4.2: Offshore wind farm and selected turbines 66
Figure 4.3: Time segmentation of the dataset 66
Figure 4.4: Approximate location of accelerometers and reference frame definition 67
Figure 4.5: Representation of the actuator disc generating a pressure discontinuity for power extraction 68
Figure 4.6: Power Coefficient Cp as a function of the axial induction factor 69
Figure 4.7: Example of a MISO regression model with two predictors 73
Figure 4.8: Example of an ε-insensitive band for a one-dimensional linear regression problem 77
Figure 4.9: Illustration of the univariate anomaly detection process using a single SVR model 81
Figure 4.10: Multivariate anomaly detection approach using four trained SVR models 82
Figure 4.11: One-class SVM classification scheme. The origin is treated as the sole representative of the negative class 83
Figure 4.12: Left-Three Gaussian distribution are shown, each characterised by a different mean but sharing a common covariance structure. The dashed lines denote the optimal pairwise decision surfaces derived from Bayes theory, while the solid lines represent the overall multi-class separation regions. Right-A sample of 30 observations per distribution is presented, alongside the linear boundaries estimated through LDA 86
Figure 4.13: Comparison of classification boundaries generated by two different approaches. The plot on the left shows the result of applying LDA after expanding the input features through a quadratic transformation, leading to non-linear decision boundaries. The plot on the right illustrates the decision surfaces obtained with QDA, which naturally accommodates class-specific covariance structures. In both cases, the separation between the three classes is evident, although QDA provides inherently curved boundaries without the need for feature expansion 87
Figure 4.14: Confusion matrix representation 88
Figure 4.15: Example of a ROC curve comparing the performance of two anomaly detectors, labelled A (blue curve) and B (green curve). The plot illustrates the trade-off between true positive rate and false positive rate across different threshold values, highlighting the relative detection capabilities of each model 90
Figure 5.1: Acceleration vs wind speed and ambient temperature (top), and QQ-plot of residuals (bottom) 94
Figure 5.2: Acceleration vs power output and wind speed, QQ-plot residuals 94
Figure 5.3: Acceleration vs wind speed and gearbox oil temperature, and QQ-plot residuals 95

Figure 5.4: Acceleration vs wind speed and inner ring bearing temperature, and QQ-plot residuals 95
Figure 5.5: Acceleration vs wind speed and pitch angle, and QQ-plot residuals 96
Figure 5.6: Acceleration vs wind speed and inner ring bearing temperature. Red circle indicate discontinuity caused by the presence of outliers in the dataset 97
Figure 5.7: Power output vs wind speed and ambient temperature 98
Figure 5.8: Cumulative explained variance and number of principal components. The red marker indicates the selected 90% threshold 101
Figure 5.9: Top: predicted vs measured nacelle vibration. Bottom: residual distribution confirming near-normal behaviour 103
Figure 5.10: Top: time-domain predicted and measured vibration values. Bottom: residuals over time. The missing data are caused by the application of label filtering 103
Figure 5.11: Validation results. Top Plot: Predicted value and Measured value representation. Bottom Plot: Residuals QQ-plot 104
Figure 5.12: Validation results. Top Plot: Predicted and measured values plotted in the time domain. Bottom Plot: Residuals time series plot. Missing data are caused by the application of label filtering 104
Figure 5.13: Test results. The first plot shows the predicted vibration values against the measured ones. Red markers are used to highlight data points associated with fault conditions 106
Figure 5.14: Test results. Time-domain representation of predicted, measured (top), and residual values (bottom). Faulty intervals are highlighted in red across different time windows 107
Figure 5.15: The first plot compares the predicted output with the measured outputs across training, validation, and test datasets, where the predicted values are shown in red. The second plot represents the residuals computed in the time domain, while the third plot the statistical distribution of these residuals. Residuals exceeding predefined thresholds, derived from the training distribution, are highlighted in black and classified as potential damaged points. 108
Figure 5.16: The second and third plots represent the residuals computed from the regression model in the time domain. Red dots indicate the true (labelled) damage points, while black dots correspond to the detected anomalies based on residual thresholding 110
Figure 5.17: ROC curve. The red circle indicates the optimal operating point selected based on Youden's index (the "knee" of the curve). The blue circle represents the threshold defined from the residual distribution of the training set 111
Figure 5.18: Predicted power output obtained using a secondary SVR model. In the bottom plot, black dots are referred to anomalies classified by the model, for data points that exceed the upper and lower thresholds 113

Figure 5.19: ROC-curve obtained from the SVR model trained on power output residuals 114
Figure 5.20: Comparison between predicted and measured power output responses in the time domain using the linear regression model 116
Figure 5.21: ROC-curve representation for the linear regression model. The red marker indicates the optimal threshold, while the blue marker represents the threshold derived from the residual distribution during training 117
Figure 5.22: Mahalanobis distance plot. Red markers denote the labelled fault data corresponding to the true failure period of WT35 120
Figure 5.23: ROC curve computed from Mahalanobis distance anomaly index 121
Figure 5.24: ROC curve computed from One-class SVM score 121
Figure 5.25: ROC-curve obtained using Mahalanobis distance based on the residuals of the Multivariate SVR model 123
Figure 5.26: ROC-curve obtained using the OCSVM with a linear kernel 123
Figure 5.27: ROC-curve obtained using the OCSVM with a Gaussian kernel. The AUC value below 0.5 indicates that the model fails to correctly distinguish between normal and anomalous instances, possibly assigning higher anomaly scores to healthy data 124
Figure 5.28: Classification scheme 127
Figure 5.29: Visual representation of the 10-fold cross-validation applied to the residual dataset used for training and evaluating the classification models 127

List of Tables

Table 1.1: Types and causes of faults in wind turbines 14
Table 3.1: Wind farms overview 52
Table 3.2: Description of status-ID labels 53
Table 3.3: Total number of features per wind farm 59
Table 4.1: Representation of wind turbines considered during analysis 65
Table 4.2: Time periods for each phase 66
Table 4.3: Values of SVR model parameters used during analysis 79
Table 4.4: Generic confusion matrix for multi-class classification, where the class k=0 corresponds to the healthy or reference state, while classes k=1… N represent different categories of damage 90
Table 5.1: Summary of regression performance metrics (target: mean nacelle vibration signal) 96
Table 5.2: Physical parameters used in SVR 99
Table 5.3: Coefficients assumed for aerodynamic power computation (Betz law) 100
Table 5.4: Performance metrics results concerning training and validation phases 105
Table 5.5: Performance metrics: Test phase results 107
Table 5.6: Performance metrics for different threshold values applied to residuals 111
Table 5.7: Performance metrics of the anomaly detection model 115
Table 5.8: Performance metrics of the linear regression model using the Betz-based formulation 116
Table 5.9: Performance metrics of the anomaly detection model based on third-degree polynomial regression 118
Table 5.10: Performance metrics-Multivariate anomaly detection models 122
Table 5.11: Performance metrics-Multivariate analysis results 125
Table 5.12: Overview wind turbines under analysis and classes identification 127
Table 5.13: Confusion matrix structure 128
Table 5.14: Confusion matrix result starting from QDA 128
Table 5.15: Confusion matrix result starting from LDA 128
Table 5.16: Performance Metrics-Classification results 128

Nomenclature

Symbol	Description
a	Axial induction factor
A	Actuator disc area
ACC	Accuracy
AUC	Area under the curve
C	Box constraint or regularisation parameter
CER	Class error rate
c_0	Upstream wind speed
C_p	Power coefficient
c_u	Downstream wind speed
D	Mahalanobis distance
$f(x)$	Decision function
F_β	F_β-score
G	Gram matrix (kernel matrix)
$\mathcal{H}$	Hilbert space
$k(x, x')$	kernel function
$\mathcal{L}$	Lagrangian function
$\dot{M}$	Mass flow rate
MAD	Median absolute deviation
MAE	Mean absolute error
max	Maximum
min	Minimum
P	Precision
$p(x)$	Probability density function
P_t	Electric power output
r	Residual
R	Coefficient of determination
REC	Recall
RMS	Root mean square value
$RMSE$	Root mean square error
R_{xx}	Autocorrelation function
s	Solidity
S	Least squares error
V	Eigenvector
Var, σ^2	Variance

w	Support vector weight
$x(k)$	Discrete random variable
X	Input variable
y	Measured output variable
$\hat{y}$	Predicted output variable
$\alpha_i^{(*)}$	Dual SVR variable
β	Regression coefficient
γ	Kernel scale parameter
$\delta_i(x)$	Discriminant function for class i
ε	Regression error
λ_0	Tip-speed ratio
λ	Eigenvalue
μ_x	Average of x
ν_i	Prior probability of class i
π_j	Explained variance ratio
ρ	Air density
σ_x	Standard deviation of x
Σ	Covariance matrix
$\phi(x)$	Mapping function

Acknowledgements

I would like to express my sincere gratitude to my supervisors for their invaluable guidance and continuous support throughout the development of this thesis, particularly for introducing me to this special subject and for believing in my work, especially regarding the presentation at the SURVISHNO conference. It was a memorable moment of my personal professional growth, that I will always cherish with heartfelt joy.

I am sincerely grateful to all my friends and colleagues for their precious advice during the entire period, especially to Mohamed, Tommaso, Fabio, Lorenzo, Andrea and Luca. The countless afternoons spent studying together will always remain a special gift that I will fondly remember.

I am deeply grateful to my mom and dad for giving me the opportunity to study at the Polytechnique of Turin, for their patience, and for teaching me to view studying as a valuable gift. A special thanks goes to my brother Simone for encouraging me during the final stage of my academic journey, without whom this phase would not have been the same.

Finally, I want to express my deepest gratitude to my girlfriend Giulia for all the support and love I needed. You have been a constant source of inspiration, a true model of resilience and determination. Thank you for always encouraging me whenever I needed it.

1. Introduction

1.1 Overview

Technological advancements and the increasingly pressing energy transition have led scientific and engineering communities to search for and develop new solutions. Nowadays, due to growing awareness of climate change, newly developed technologies are increasingly based on renewable energy sources rather than fossil fuels. Greater attention is now paid to electricity production through plants composed of numerous wind turbines, photovoltaic panels, hydroelectric turbines, and nuclear reactors. Meanwhile, progress in electronics and information technology has provided new tools to ensure consistently high performance in the production of energy, along with continuous monitoring to reduce unplanned downtime and its associated dead time.

In this context, the introduction of Artificial Intelligence (AI) algorithms and widespread use of Machine Learning (ML) techniques have significantly improved the accuracy of predictive maintenance systems, especially for Early Fault Detection, enabling the identification of damage and fault conditions before complete system failure occurs.

Today, a wind turbine is a complex electromechanical system that enables the conversion of wind energy into electrical energy. Its complexity arises from the many components working together to achieve energy generation. Therefore, understanding and detecting anomalies related to any of these components is a challenging task. In practice, each component is prone to malfunction or failure due to random events or ageing-related degradation, which may cause system interruptions and reduced efficiency. As a result, unexpected abnormal behaviour may be detected and subsequently classified as faults or failures.

According to the Global Wind Energy Council, operation and maintenance costs can account for 10% to 20% of the total electricity production cost for onshore turbines, and up to 20% to 25% for offshore wind turbines. As such, the wind energy industry has a strong incentive to enhance the reliability, safety, availability, and productivity of wind turbine systems.

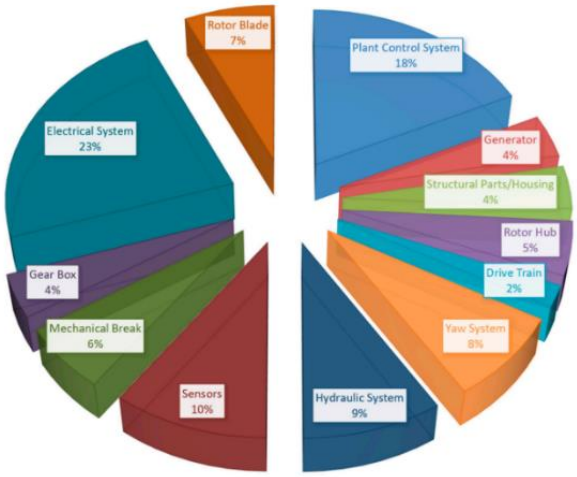

Figure 1.1: Percentage distribution of common failure modes in primary wind turbine components

The pie chart (Fig. 1.1) illustrates the distribution of typical fault types in wind turbines. The main causes of these faults are summarised in Table 1.1. If an unexpected fault is not detected in its early stages and no corrective action is taken, it may lead to further, more severe failures.

Table 1.1: Types and causes of faults in wind turbines

Types of Faults	Causes of Faults
Faults on blades and rotors	Corrosion of blades and hub; crack; reduced stiffness; increased surface roughness; deformation of the blades; errors of pitch angle; and imbalance of rotors, etc.
Faults on gearbox	Imbalance and misalignment of shaft; damage of shaft, bearing and gear; broken shaft; high oil temperature; leaking oil; and poor lubrication, etc.
Faults on generator	Excessive vibrations of generator; overheating of generator and bearing; abnormal noises; and insulation damage, etc.
Faults on bearing	Overheating; and premature wear caused by unpredictable stress, etc.
Faults on main shaft	Misalignment; crack; corrosion; and coupling failure, etc.
Hydraulic faults	Sliding valve blockage; oil leakage, etc.
Faults on mechanical braking system	Hydraulic failures; and wind speed exceeding the limit, etc.
Faults on tower	Poor quality control during the manufacturing process; improper installation and loading; harsh environment, etc.
Faults on electrical systems/devices	Broken buried metal lines; corrosion or crack of traces; board delamination; component misalignment; electrical leaks; and cold-solder joints, etc.
Faults on sensors	Malfunction or physical failure of a sensor; malfunction of hardware or the communication link; and error of data processing or communication software, etc.

Condition monitoring systems play a crucial role in assessing the operational health of wind turbines. They aim to confirm whether the system is working properly, detect early signs of faults or malfunctions, identify the components involved, and assess the severity of any anomalies. This enables timely corrective actions, potentially avoiding further degradation and ensuring operational safety. In parallel, prognostic techniques aim to forecast future failures and estimate the remaining useful life (RUL) of components, thus allowing for the planning of maintenance interventions in advance.

Over the past decades, a wide range of monitoring approaches have been developed in both academic research and industrial applications. These typically rely on continuous data acquisition using dedicated sensors, followed by numerical analysis through signal processing or machine learning algorithms. A key requirement for any effective monitoring strategy is the selection of physical quantities that are highly sensitive to mechanical or operational changes, particularly those that occur when a fault arises.

Among the various techniques available, vibration analysis is widely appreciated for its ability to detect faults with high accuracy. In mechanical systems, normal operation produces distinct vibration patterns, and any deviation from these patterns could be a signal of emerging faults. This characteristic makes vibration analysis an invaluable tool for diagnosing system issues.

Other condition monitoring methods include:

- Acoustic Monitoring: Variations in the sound emitted during operation or testing, such as those generated by a hammer impact, can highlight structural defects. Airborne sound signals are captured by microphones and pressure sensors, which are then processed to locate and analyse the defect.
- Acoustic emission: This involves detecting high-frequency elastic waves generated by crack growth or plastic deformation within materials. While this method is capable of early damage detection, its application is limited by the complexity of sensor placement and signal interpretation.
- Oil debris analysis: Mechanical wear can cause particles to detach and circulate within the lubricant. These particles, analysed in terms of quantity, size, shape, and composition, can offer insights into internal damage. However, this method is not predictive, as it only detects damage after material separation has occurred.
- Performance Analysis: Access to detailed performance data, such as power output or efficiency, allows for the evaluation of the system's condition. For instance, a sudden drop in power output could indicate the onset of a fault. Unfortunately, such data is not always accessible or sufficiently detailed.
- Thermal Monitoring (Thermography): Temperature measurements can reveal abnormal heat generation associated with friction, especially in components like bearings nearing the end of their life cycle. Nonetheless, this method often fails to detect early-stage faults.

Among these methods, vibration analysis provides an optimal combination of cost-effectiveness, sensitivity, and ease of application. It causes little disruption to turbine operations while offering fast detection of mechanical irregularities. This supports a shift from traditional time-based maintenance (preventive) to condition-based maintenance (predictive), which improves reliability, reduces costs, and minimises downtime.

1.2 Comparative Maintenance Strategies

Based on the application, several different maintenance strategies can be adopted. The main techniques are briefly summarised below:

- Run-to-break: This is one of the most traditional methods, in which the machine continues to operate until it fails. This strategy results in the longest time between shutdowns, but when failure occurs, it is commonly catastrophic. The time required to repair the machine can be significantly extended, particularly when replacing damaged components, which involves design and production stages.
- Time-based Preventive Maintenance: In this approach, maintenance is carried out at periodic intervals that are shorter than the expected 'time-between-failures'. Generally, maintenance is scheduled at intervals when 1-2% of machines are expected to fail. This method offers several advantages, such as the ability to plan maintenance in advance, thereby reducing the risk of catastrophic failure. However, its drawbacks include the possibility of unforeseen failures and excessive maintenance, which leads to unnecessary component replacements. Time-based preventive maintenance is suitable when the failure time can be estimated with reasonably accuracy. For instance, certain components experience wear or fatigue at a consistent rate, while others, like rolling element bearing show significant variation in their performance, resulting estimates where the average time to failure may be two or three times longer than the minimum predicted value.
- Condition-based Maintenance (CBM): Also known as 'Predictive Maintenance', this strategy involves predicting machine breakdowns through regular condition monitoring. Maintenance is then performed at the optimal time, which offers clear advantages over both run-to-break or preventive maintenance. However, it requires reliable condition monitoring techniques, capable not only of determining the current condition of the machine but also predicting the RUL. Initially, CBM achieved the best results in industries where machines were required to operate continuously for extended periods, such as in power generation. In these industries, machines typically run at a nearly constant speed and under stable loads, which reduces the technical challenges related to condition monitoring. However, with the advent of more advanced diagnostic techniques, condition monitoring has expanded to industries where machines experience more variable speeds and loads. For example, wind turbines are equipped with a gearbox that allows for adjusting the rotational speed, starting from the low rotor speed designed to avoid high centrifugal forces, and scaling up to the mains frequency.

As a consequence, CBM is the most widely used maintenance strategy, and its purpose is to prevent failures. This can be achieved through the application of diagnostic techniques that detect the presence of damage, classify its severity, and provide an estimate of the RUL, thereby improving the scheduling of the maintenance operations.
A typical CBM program includes three main steps, as shown in Fig. 1.2:

Figure 1.2: General workflow adopted in a CBM program

The data acquisition step consists of collecting fundamental information from a healthy system. Typically, this data contains noise, which makes a preprocessing step necessary. This step involves data cleaning and the application of normalisation techniques. Next, the data processing phase handles the interpretation and management of the collected data. Finally, in the maintenance decision-making step, actions are determined based on the results of the data analysis.

1.3 Diagnostic Definition

In the context of CBM, diagnostics and prognostics are two fundamental pillars. Diagnostics refers to the process of fault detection, isolation and identification. Specifically, detection involves recognising that something is wrong during system monitoring; isolation focuses on localising the faulty component, and identification determines the nature and severity of the fault. Conversely, prognostic aims to predict the occurrence of faults before they manifest. Consequently, diagnostic actions are typically triggered when prognostic models fail or when significant uncertainty exists. Maintenance decisions are then based on the health information derived from the processing of acquired data.
It is important to distinguish fault and failure, which are closely related concepts. The detection of unexpected or ‘abnormal’ behaviour must be classified into one of these two. A fault is defined as an unacceptable deviation of a system or structural parameter from normal operating conditions. Failure, on the other hand, occurs when a component or system can no longer fulfil its intended function. Both phenomena often result from

progressive degradation, such as damage or defects. In particular, according to the definitions established by Worden and Dulieu-Barton (2004):

- Fault: A condition in which the structure is no longer able to function properly. If quality is defined as the structure's ability to fulfil its intended purpose, then a fault can be interpreted as an unacceptable reduction in quality.
- Damage: The structure operates below satisfactory levels and deviates significantly from its ideal state.
- Defect: Related to material properties. All materials, by their nature, contain a statistical distribution of defects, classified in material science as point, linear, surface, or volumetric defects.

These terms are inherently linked: a defect may lead to damage, which can in turn cause a fault. Moreover, the presence of any of these-defect, damage, or fault-can ultimately lead to failure. They can often be detected at an early stage because they alter the dynamic characteristic of the system. The amplitude of such changes can serve as an indicator for inspection and control. Since diagnostics aims to trace the root cause of malfunction from observable effects, vibration measurements are an effective tool for monitoring the health of wind turbines.

However, uncertainty always exists due to confounding factors-external influences that may also affect a system's dynamic response. Two key considerations must be kept in mind:

- No sensor is capable of directly measuring damage; rather, only its consequences or manifestations on the system's behaviour can be detected indirectly through measurable quantities.
- Alterations in the dynamic characteristic of a system may indeed result from the presence of damage but can also be induced by other external or operational factors unrelated to structural degradation.

Wind turbines are particularly susceptible to strong dynamic effects from their environment, such as temperature, wind speed, humidity, pressure, etc. (Fig. 1.3), which can affect material properties and, consequently, the turbine's dynamic response. This highlights the need for methods capable of filtering out environmental influences to isolate changes attributable to actual damage.

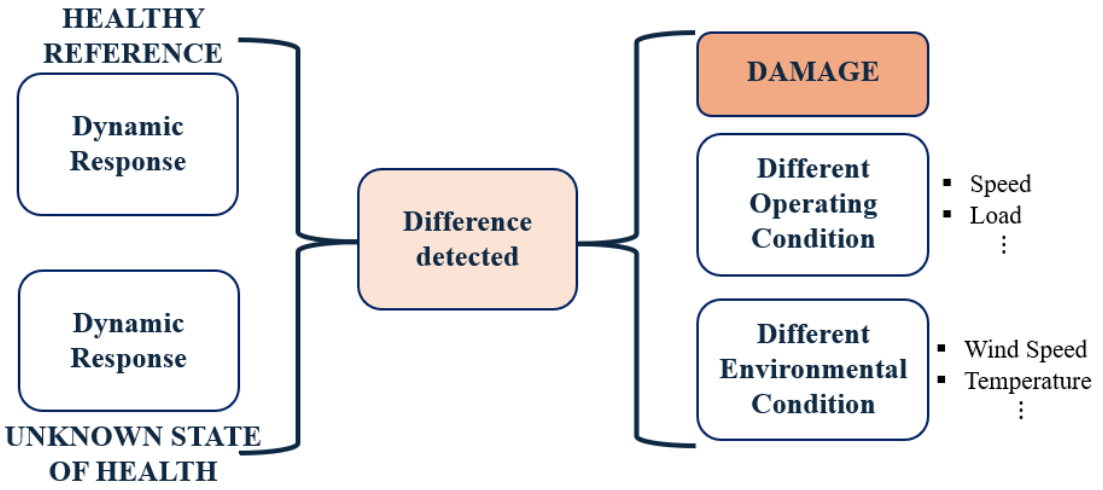

Figure 1.3: Schematic representation of a damage detection process based on dynamic response comparison

Furthermore, diagnostic extracts information from experimental data collected through dedicated sensors. The accuracy of diagnostic analysis depends largely on the quality of the collected data. Crucially, the knowledge acquired through condition monitoring follows a hierarchical structure, first proposed for first by Rytter in 1933, and later refined by Worden and Dulieu-Barton in 2004:

- Detection: A qualitative indication that damage exists
- Localisation: Estimating the probable location of the damage
- Classification: Identifying the type of damage
- Assessment: Quantifying the severity or extent of the damage
- Prediction: Estimating the remaining safe and useful life

The first four levels fall under diagnostics definition, while the last one is linked to the prognostics. The extraction of meaningful knowledge from experimental data in condition monitoring typically follows two main approaches:

- Model-based approach: This method is applicable when a solid theoretical understanding of the system exists, allowing the development of a mathematical or physical model. In this context, damage identification is formulated as an inverse problem, where the model parameters are continuously updated using newly acquired data. Deviations from nominal parameter values can thus be interpreted as indicators of structural damage.
- Data-based (or data-driven) approach: When prior knowledge about the system is limited or the system under investigation is highly complex, such as in the case of wind turbines, relying on purely analytical models becomes impractical. In such scenarios, data-driven methods are employed. These techniques leverage pattern recognition and machine learning algorithms to identify underlying regularities in

the data, enabling classification and detection of abnormal conditions without explicit physical modelling.

Classic examples of model-based vibration analysis include Single-Degree-Of-Freedom (SDOF) and Multi-Degree-Of-Freedom (MDOF) systems (Fig. 1.4), which are simplified yet effective representations of structural dynamics.

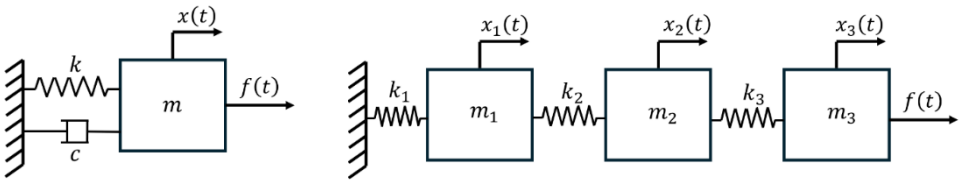

Figure 1.4: Schematic representation of a SDOF system (left) and a MDOF system (right)

In this study, the focus is placed on the data-driven approach, which is particularly suitable for analysing complex systems through vibration-based monitoring techniques. This strategy allows for the identification of early signs of damage by processing sensor-acquired data, even in the absence of comprehensive physical models.

1.4 Vibration Monitoring

Vibration monitoring represents a specific form of condition monitoring that exploits vibrational responses to assess the health status of the machine under investigation and, to detect anomalies potentially associated with damage. Vibration is inherently a dynamic phenomenon which, in conservative systems, arises from the continuous exchange between potential and kinetic energy. Over the past decades, vibration analysis has emerged as one of the most widely adopted techniques in condition monitoring, primarily because each mechanical component exhibits a distinct vibrational signature, that can be continuously observed. Deviations from this baseline signature may be indicative of the onset of damage. The literature on vibration-based monitoring in wind turbines is extensive, given that vibrational signals are highly sensitive indicators to mechanical degradations and faults.

In general, all machines, including wind turbines, generate vibrations during normal operations. However, wind turbines are typically installed in geographically exposed areas where environmental conditions are highly variable and often sever. As a result, these systems are subjected to a wide spectrum of loads, especially dynamic loads, which can induce complex vibrational behaviour. Moreover, the structural flexibility of wind turbines, coupled with their operation in inherently unsteady environments, often leads to significant and persistent vibrational responses.

As highlighted in the previous sections, vibration monitoring offers several advantages compared to alternative diagnostic approaches. For instance, unlike oil analysis, which involves delays due to sampling and laboratory processing, vibration analysis provides immediate insights into machine behaviour. Thermography techniques, although widely used in contexts such as fault detection in railway vehicle bearings [33], are generally less effective for early-stage fault detection in rotating machinery, since temperature anomalies tend to manifest only in the later stages of failure. Similarly, acoustic emission monitoring may necessitate the acquisition of large datasets to capture sporadic burst events associated with incipient damage.

Wind turbines generate mechanical forces and motions that give rise to vibrations, electrical signals, and thermal outputs. These responses justify modelling them as mechanical systems with multiple excitation sources and multiple observable outputs, thus conforming to a Multiple-Input and Multiple-Output (MIMO) system framework (Fig. 1.5).

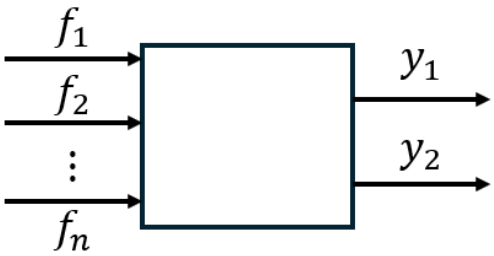

Figure 1.5: MIMO system

A central challenge in vibration-based damage detection lies in determining whether the observed deviations originate from changes in excitation conditions or from internal alterations within the system itself. Accurately establishing this relationship is essential to effectively link machine behaviour to damage indicators. Despite the inherent complexity of this task in real-world systems, this work adopts vibration monitoring as the primary diagnostic technique for the analysis of nacelle vibrations in a wind farm located in northern Germany. This choice is motivated by the relative simplicity and cost-effectiveness of vibration data acquisition compared to other condition monitoring approaches.

1.5 Anomaly Detection

Anomaly detection plays a crucial role in enabling the maintenance of wind turbines. It refers to the process of identifying patterns in data that deviate from expected behaviour. These ‘abnormal’ values are commonly called anomalies, outliers, discordant

observations, exceptions, peculiarities or contaminants depending on the application domain. The concept of detecting outliers in data was first studied in the statistical field as early as the 19th century (Edgeworth, 1887). Since then, numerous techniques have been developed across various research communities, particularly for the condition-based maintenance of industrial machinery health.

Among the most widely adopted approaches are SCADA-based solutions, especially those that incorporate vibration measurements. These methods are favoured due to their ability to generate large volumes of data without incurring additional sensor costs.

SCADA system acquires high-frequency vibration signals from critical components, such as the gearbox, nacelle or tower turbine. This data allows for the early detection of fault conditions, often before actual failure occurs. Moreover, the large volume of data produced necessitates the application of machine learning (ML) techniques. ML algorithms tend to perform better with larger datasets, enhancing their ability to distinguish between normal and anomalous operating conditions.

In scientific literature, the terms anomaly detection and novelty detection are often used interchangeably, as the underlying methodologies and algorithms employed in both contexts share substantial similarities. Nonetheless, a conceptual distinction can be made between the two. Anomaly detection is generally concerned with identifying deviations or irregularities in data that suggest the presence of known or expected faults. In contrast, novelty detection aims to uncover previously unseen or rare events that differ from both normal behaviour and any known fault conditions. In novelty detection, the system learns a model of normality by training on a lar gest of data samples that represent only the normal (positive) class. New observations are them evaluated by comparing them against this learned model, and a novelty score is computed to quantify the degree of deviation. This score, which may be probabilistic or non-probabilistic in nature, is compared to a predefined decision threshold. Observation that exceeds this threshold are classified as novel or abnormal, indicating that they significantly diverge from the established notion of normality. By contrast, in anomaly detection, the focus is primarily on detecting irregularities within operational data that are assumed to be mostly normal. These anomalies can take the form of sudden transient, outliers, or subtle deviations that may not correspond to a previously defined fault pattern. In the context of rotating machinery or wind turbines, such anomalies are particularly important as they may indicate the early onset of mechanical damage misalignments, or other forms of degradation that require timely intervention.

In addition, a fundamental aspect of anomaly detection is understanding the nature of the anomalies being investigated. Typically, anomalies can be categorised into three main types:

- Point Anomalies: This is the simplest and most extensively studied type of anomaly. A point anomaly occurs when a single data instance deviates

significantly from the rest of the data and is therefore considered anomalous (Fig. 1.6).

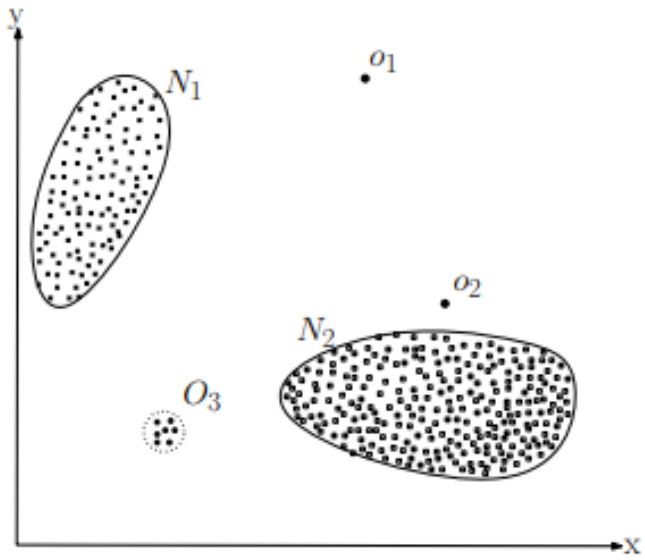

Figure 1.6: Point anomalies examples in a 2-dimensional space: O1, O2 and O3 are isolated observations that significantly deviate from the normal data distributions N1 and N2

- Contextual Anomalies: Also known as conditional anomalies, these occur when a data instance is anomalous in a specific context but may be normal in another. The anomaly is defined by the value of certain behavioural attributes within a particular context. For example, in a time series of monthly temperatures, a temperature value of 5 °C in winter (at time t_1) may be normal, whereas the same value during summer (at time t_2) would be considered anomalous. Contextual anomalies are commonly studied in time series and spatial datasets.
- Collective Anomalies: In this case, a group of related data instances is considered anomalous when viewed together, although individual instances in the group may appear normal. An example is shown in Fig. 1.7, where a wind turbine's power output remains at a consistently low value for an unusually long period, forming a collective anomaly.

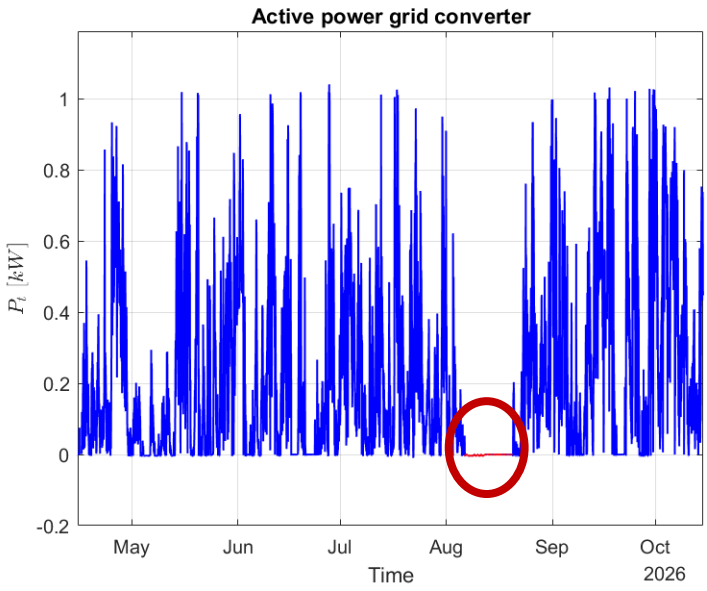

Figure 1.7: Collective anomalies example in wind turbine's power output. In red is highlighted the anomaly

It is important to note that point anomalies can appear in any dataset, while collective anomalies require interrelated data, such as sequence or temporal series. Contextual anomalies, on the other hand, depend on the presence of contextual information within the data. Moreover, both point and collective anomalies can also be considered contextual when analysed within a specific context. Therefore, by integrating contextual attributes, problems involving point or collective anomalies can be reframed as contextual anomaly detection tasks.

Another key aspect of any anomaly detection technique is how the detected anomalies are presented. Typically, the output falls into one of two categories:

- Scores: These methods assign an anomaly score to each test instance, reflecting the degree of abnormality. The result is a ranked list of potential anomalies, allowing domain experts to apply a threshold for identifying the most relevant ones.
- Labels: In this case, each instance is classified as either normal or anomalous. While these binary methods do not provide a flexible way to tune sensitivity, the choice of internal parameters in the model can influence the decision boundary.

1.6 Machine Learning

The application of anomaly detection techniques to wind turbines can be implemented through various methodologies. In recent years, research has demonstrated that

disciplines such as statistics, data mining or machine learning are capable of delivering highly accurate and effective results. This thesis focuses on the application of supervised machine learning methods aimed at detecting anomalies potentially associated with fault conditions. To fully understand the adopted workflow, an introductory overview of machine learning concepts is necessary.

ML is a branch of AI concerned with developing algorithms that can automatically extracts knowledge from data with minimal human intervention. To core objective of ML is to devise general-purpose techniques capable of being applied across different domains. These methods are fundamentally based on three key components:

- Data: ML algorithms are designed to extract meaningful patterns from data without requiring explicit domain-specific knowledge.
- Model: A model refers to the algorithm or mathematical structure that processes the input data. A well-designed model is expected to generalise effectively to previously unseen data, enabling robust pattern recognition.
- Learning: Learning denotes the process of optimising model parameters to discover inherent patterns in the data. When this optimisation leads to enhanced performance on a specific task, the model is said to have 'learned'.

In the ML context, the term algorithm can refer to two distinct but interconnected functionalities:

- As a predictor, it makes estimations or decisions based on input data.
- As a trainer, it adjusts its internal parameters in response to data, thereby improving its performance, this process is known as training.

Training a model entails optimising its parameters based on a given dataset and an associated utility or loss function, which evaluates the accuracy or effectiveness of the model's predictions.

Machine learning method are traditionally grouped into three broad categories, as illustrated in Fig. 1.8.

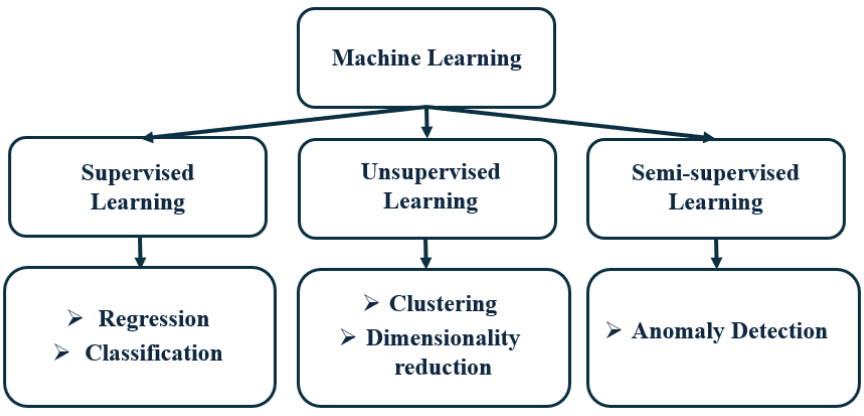

Figure 1.8: Machine learning taxonomy

Supervised learning aims to predict one or more output variables, also referred to as response variable, denoted by $\boldsymbol{Y} = (y_1, y_2, \dots, y_N)$, from a corresponding set of input or predictor variables $\boldsymbol{X} = (x_1, x_2, \dots, x_N)$. These predictions rely on a training dataset composed of input-output pairs $(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$, where the relationship between variables is fully known. The goal is to determine a function $f: \boldsymbol{X} \rightarrow \boldsymbol{Y}$ that accurately maps the inputs to the outputs, minimising the prediction error on unseen data. In contrast, unsupervised learning deals with input data that lacks labelled outputs. Instead, the focus shifts to discovering hidden patterns or structures in the data, typically through techniques such as clustering or dimensionality reduction.
Moreover, while supervised learning requires a significant amount of labelled data to be effective, often making the process expensive and time-consuming, unsupervised learning offers a more flexible alternative but may suffer from reduced accuracy, especially with previously unseen data. To bridge this gap, semi-supervised learning combines both approaches by using a small, labelled dataset along with a larger set of unlabelled data. This allows for improved model performance without the full cost of manual labelling.
Semi-Supervised learning can be further divided into two categories: semi-supervised classification and semi-supervised clustering. In the field of anomaly detection, the task is often tackled as semi-supervised classification problem. Rather than using a traditional two-class setup, the focus is place on one-class classification, which aims to detect deviations from normal behaviour. These deviations can indicate novel or unexpected patterns in the data, potentially associated with system faults or failures.

1.6.1 Regression Methods

Regression is a Supervised Learning technique used to determine the best correlation between predictors variables (input training data) and predicted values (output data). More specifically, this approach involves defining a function f that maps input data, based on a set of training inputs x_i, to a corresponding output value $f(x) \in \mathbb{R}$, representing the predicted value. These outputs correspond to noisy observations $y_i = f(x) + \varepsilon$, where ε is a random variable accounting for noise in the signal.
This function aims ono tony to accurately predict the training data but also to generalise effectively to new, unseen data. First, a regression involves several key steps, as indicated in the following:

- Model selection and parametrisation: For a given dataset, a suitable function that must be chosen from a class of candidate functions, such as a polynomial function.

The choice includes parameterisation, such as the polynomial degree, to enhance prediction accuracy. In addition, selecting an appropriate model also facilitates comparisons among alternatives to ensure the simplest explanation of the training data.

- Parameter estimation: Once the model regression is selected, a loss (or objective) function is defined to quantify the fit. The model parameters are then estimated by minimising this function.
- Overfitting: A common issue in regression, overfitting occurs when the model fits the training data too closely and fails to generalise. It often arises when the model is too complex or overly parameterised relative to the data.
- Modelling uncertainty: Because the dataset represents only a finite sample of possible inputs, the model cannot account for all scenarios. As a result, uncertainty modelling is needed at test time to provide confidence measures for predictions.

Linear Regression, one of the simplest and most widely used algorithms in statistics, aims to identify the best-fit line between a single into and single output (base problem). Based on the least squares method, it assumes a linear parametric form and estimates parameters by minimising the sum of squared residuals. This method is computationally efficient and performs well when the relationship between variables is approximately linear. However, it struggles with complex non-linear relationships, resulting in suboptimal fits when its assumptions do not hold.

In many practical scenarios, the response variable may depend on more than one predictor, making simple SISO (Single-Input and Single-Output) regression inadequate. This leads to the adoption of more general models, such as:

- MIMO Regression: Multiple-Input and Multiple-Output
- MISO Regression: Multiple-Input and Single-Output

These models share similar assumptions to SISO linear regression, including the following key assumptions:

- Linearity between inputs and outputs
- Multicollinearity lack: Multicollinearity is present when the independent variables are not independent of each other
- Normal multivariate: Residuals are normally distributed

Support Vector Regression (SVR) is the extension of the Support Vector Machine (SVM) to regression, as was proposed by Vapnik and his colleagues in 1992. While linear regression is usually a parametric model, SVR does not assume a specific parametric form but instead relies on kernel functions to project input data into a higher-dimensional feature space, where a linear function is then fitted the kernel function represents an inner product in this feature space:

$$k(x, x') = \langle \phi(x), \phi(x') \rangle$$

Here, ϕ is a mapping from the input space, represented by vectors x and x', to a Hilber space $\mathcal{H}$, commonly referred to as the feature space. The kernel trick allows these computations to be performed implicitly in the feature space without explicitly computing $\phi(x)$.

SVR uses these kernel functions to learn either linear or non linear mappings, while its capacity is controlled by parameters that are independent of the dimensionality of the feature space. Similar to SVM classification, the SVR algorithm minimises a convex loss function, producing a sparse solution. A crucial aspect of SVR is the use of an ε-intensive loss function, which ignores small errors (within a defined threshold), focusing only on significant deviations.

As a result, SVR offers a flexible and robust alternative to traditional regression methods, particularly when dealing with non-linear relationships or limited labelled data.

1.6.2 Dimensionality Reduction

Dimensionality reduction is a fundamental technique within the scope of unsupervised learning, particularly relevant when analysing high-dimensional datasets comprising a large number of features. While this abundance of information may enhance data richness, it simultaneously increases analytical complexity and the risk of including redundant or irrelevant information, such as duplicated or correlated features.

To address these challenges, dimensional reduction serves as an effective strategy to eliminate noise and redundancy, thereby improving both the efficiency and the interpretability of subsequent analysis. This process is typically carried out in two stages: feature selection and feature extraction. The core idea behind dimensionality reduction is to project data samples from a high-dimensional space into a lower-dimensional space while preserving the essential structure and information. In mathematical terms, given an input vector $\boldsymbol{X} = \{x_1, x_2, \dots, x_n\}^T$, the main purpose is to map it into a lower representation $\boldsymbol{Y} = \{y_1, y_2, .., y_m\}^T$, through the following relation $\boldsymbol{Y} = f(\boldsymbol{X})$.

One of the most widely used techniques for linear dimensionality reduction is Principal Component Analysis (PCA). Originally introduced by Pearson (1901) for bivariate problems and later extended by Hotelling (1933) for multivariate datasets. PCA identifies the directions, i.e. principal components, along which the data exhibit the greatest variance. In the context of high-dimensional datasets, PCA helps to reduce dimensionality by retaining only the most informative linear combinations of the original features.

However, PCA is inherently a linear method, which limits its effectiveness when the data lie on a non-linear manifold. To overcome this limitation, kernel PCA has been proposed as an extension. By employing the kernel trick, the data are implicitly mapped into a higher-dimensional feature space where linear separation becomes feasible. In this transformed space, PCA is then performed, allowing for non-linear dimensionality reduction while still preserving the interpretability and structure of the data.

1.6.3 Anomaly Detection in Machine Learning

As has already been introduced, anomaly detection is the process of identifying unusual patterns in data that deviate from expected behaviour. A simple method typically consists of centring around the definition of a normal behaviour domain for each approach. Any data observation that falls outside this normal range is deemed an anomaly. Additionally, there can be labels associated with data instances, indicating whether it is classified as normal or anomalous. As a result, many anomaly detection methods rely on classification techniques.
Referring to Fig. 1.8, which illustrates the ML taxonomy, anomaly detection techniques follow the same classification scheme.
In supervised anomaly detection, models are trained on datasets that include both normal and anomalous labels. The goal is to learn a decision boundary that can accurately separate the two classes. In contrast, semi-supervised anomaly detection techniques are trained using only data from the normal classes. These methods are particularly useful when anomalous data is rare or unavailable, which is often the case in real-world industrial scenarios.
Unsupervised anomaly detection methods do not require labelled data. They rely on the assumption that anomalies are rare and significantly different from the majority of data. However, if this assumption does not hold, such methods can produce a high rate of false alarms. In the supervised approach, anomaly detection can be performed by training a classification model using a dataset presenting both anomalous and normal data, composed of labelled instances. During the training phase, the model (or classifier) learns to distinguish between normal and anomalous conditions based on the provided labels. In the testing phase, the trained classifier is then used to predict whether new, unseen data instances belong to the normal class or represent anomalies. In the semi-supervised is, just a dataset with normal data is needed, so no anomaly examples occur.
Two main approaches are widely used in anomaly detection: one-class and multi-class classification (Fig. 1.9).

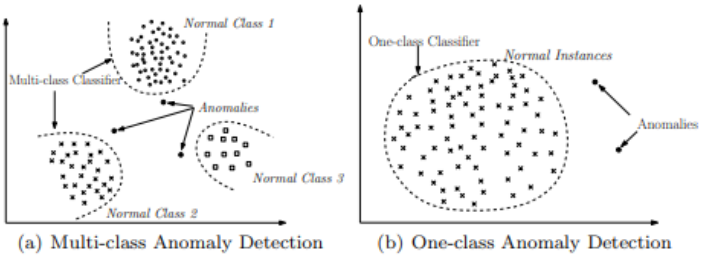

Figure 1.9: Anomaly detection based on classification approach

In one-class, the model is trained solely on normal data, assuming that all training instances belong to a single class. This approach is commonly implemented using techniques such as One-class SVM. Conversely, multi-class involves training the model on multiple classes of normal behaviour, enabling it to distinguish among several types of normal conditions.

When labelled data are not available, supervised methods cannot be applied, and an unsupervised approach must be used instead. Clustering algorithms are widely adopted in this context to group similar instances together. Anomalies are then identified as instances that do not conform to the general clustering structures. These clustering-based anomaly detection methods typically rely on one or more of the following assumptions:

- Normal instances tend to form well-defined clusters, whereas anomalies do not belong to any clusters. Methods such as DBSCAN, ROCK, and Shared Nearest Neighbour (SNN) clustering are commonly used in this case.
- Normal instances are located close to the centroid of their nearest cluster, while anomalies lie at a greater distance. Techniques like k-means clustering and expectation maximisation fall in this category.
- Normal instances belong to large, dense clusters, while anomalies are associated with small or sparse clusters. A notable example of this approach is the cluster-based local outlier factor (CBLOF).

These methods generally require the computation of distances between pairs of instances. In this respect, clustering shares similarities with nearest neighbour-based anomaly detection methods, although the latter focus on local neighbourhood structures rather than global groupings.

1.7 Challenges and Recent Research

While anomaly detection may initially appear to be a straightforward task, its application in real-world scenarios is considerably more complex. Various factors must be taken into account, highlighting the true challenges involved in its implementation:

- Defining a region that accurately represents the normal operating behaviour of the machine is inherently difficult. Furthermore, the threshold that separates normal from anomalous behaviour is often not sharply defined. As a result, numerous data points may lie near the boundary, leading to potential misclassification of normal and anomalous conditions.
- In many real-world scenarios, the definition of 'normal behaviour' is not static but varies over time due to external factors such as temperature, humidity, or pressure changes-particularly relevant in environments like wind farms. Consequently, a model trained on historical data may no longer be representative in future conditions.
- The availability of labelled data for training, validation, and especially for testing is often limited. In some cases, labels may not be accessible at all, or their accuracy may be questionable.
- Real-world data is typically affected by noise, which can closely resemble actual anomalies. Moreover, damages or faults may manifest subtly within the data, requiring robust preprocessing techniques to effectively isolate and identify them.

These challenges highlight that anomaly detection is a non-trivial task, highly dependent on the specific system under investigation and the quality and characteristics of the available data.

Currently, the research field is highly active in developing advanced algorithms and methodologies to improve detection accuracy and overcome the above-mentioned limitations. A growing number of studies propose autonomous systems based on adaptive machine learning algorithms. Among them, Transfer Learning and Deep Learning approaches are widely used due to their excellent performance in complex detection tasks [4,6,11]. These methods are capable of monitoring machine health and detecting faults at an early stage. However, they come with some drawbacks: they typically require large amounts of training data, long training times, and significant computational resources.

Other commonly used techniques include Support Vector Machine (SVM) [18,15,14], Decision Trees [16], and Artificial Neural Networks (ANN) [17]. In recent years, alternative strategies have emerged, such as physics-informed deep learning [11,18,14], convolutional neural networks (CNN) [8], and hyperparameter optimisation through a hierarchical search algorithm [9]. Autoencoder (AE) models have also gained popularity, as seen in the works of Cyriana M.A. Roelofs et al. [4] and others [32,6]. These models aim to detect as many anomalies as possible by training on carefully prepared datasets.

Due to the high computational cost of deep learning methods, some researchers have focused on developing simpler techniques that still offer good performance while being easier to implement and maintain [7]. For example, Sinvaldo R. Moreno et al. [12] proposed fault detection models based on power curve analysis. These models monitor

turbine performance and detect anomalies caused by factors such as nacelle misalignment, wind transducer displacement, or increased blade load due to sensor failure. Their approach estimates the ideal power output by considering physical parameters such as air density, rotor area, and wind speed, using machine learning models solely for health monitoring purposes.
A hybrid Genetic Algorithm (GA) classifier has been introduced by Sunil Tyagi and S. K. Panigrahi [5]. This method combines an Artificial Neural Network (ANN) classifier trained with GA-based optimisation (GA-BP) to detect gearbox faults based on vibration measurements, demonstrating the effectiveness of integrating different machine learning strategies to enhance diagnostic capability.
Other relevant contributions include the work by Castellani et al. [26], which combines Principal Component Analysis with Mahalanobis distance to detect anomalies from tower vibration measurements, showing how a novelty index can serve as a key detection tool while also reducing false alarms. In another study [25], they apply Support Vector Regression (SVR) to monitor the temperature trends of bearings for early fault diagnosis.
A notable example is the study proposed by Dhiman et al. [31], who applied various Support Vector Regression-based methods to monitor gearbox temperature, using feature selection and statistical tests on the residuals as anomaly indicators.

1.8 Thesis Objectives

The goal of this thesis is to develop an anomaly detection method in order to perform a complete diagnostic assessment of the wind turbine under investigation. One of the main objectives is to define a strategy that provides a general overview of the machine's health status. This approach should be generalisable to other turbines as well, thanks to the ability of the models considered in this thesis to generalise physical behaviour.
Such a methodology would not only help identify the presence of faults but also distinguish between different types of failures and damages, an essential requirement for timely and effective planning of predictive maintenance. This is crucial in the context of wind energy where continuous electricity production is vital to meet energy demand.
The proposed diagnostic procedure is based on the selection of a specific physical parameter, nacelle vibration, as a reference indicator. Vibrations are highly sensitive to physical variations and can be used as a baseline signal, so that any deviations from their expected profile can be analysed to determine whether they are due to actual faults or simply to variations in environmental or operating conditions (e.g., changes in wind speed or ambient temperature), i.e. confounding factors. Vibration monitoring methodologies

are widely adopted in the field of diagnostic, and in this work the analysis is primarily focused on time-domain features, which are then processed using ML models.

In summary, after discussing the main advantages and limitations of various early fault detection strategies for wind turbines, it becomes essential to develop a solid understanding of the functional behaviour of their mechanical and electrical components, as well as of power control principles. This knowledge provides the necessary physical background for defining the specific objectives of this thesis.

Moreover, the aim of this work is to integrate physical modelling insights into a machine learning framework, with the goal of improving the accuracy and robustness of predictive maintenance algorithms. In the next chapter, a concise overview of the main failure modes offers valuable context for interpreting the analytical results. Given the complexity of wind turbines as electromechanical systems, vibrations caused by mechanical faults serve as critical indicators of potential damage. Therefore, understanding the relationship between vibration patterns and fault types is fundamental to the development of effective vibration-based anomaly detection methods.

These insights serve as the foundation for the data-driven analysis presented Chapter 4, while the detailed methodological framework and proposed approach are introduced in Chapter 2.

2. Wind Turbines

In this chapter, the operating principles and structural components of wind turbines are examined, with a particular focus on their typical failure mechanisms and power control strategies. The section also outlines the most common faults observed in real-world applications, laying the foundation for the subsequent analysis of condition monitoring and anomaly detection.

2.1 Wind Turbines Overview

Wind turbines are machines that convert wind energy, which directly dependent on wind velocity, into electrical energy. While traditional windmills convert wind energy into mechanical energy, modern wind turbine acts as electrical generator connected to various types of networks, including battery-charging systems, residential microgrids, isolated power systems, and large-scale utility grids, enabling energy storage and distribution.
The energy conversion process is governed by the principles of aerodynamics. Specifically, the blade design creates a pressure differential across the blade surfaces, generating lift. This aerodynamic force induces torque on a rotating shaft, which drives a generator that transforms mechanical energy into electrical power. This makes wind turbine as a key technology for producing electricity from renewable source. However, unlike many conventional generators, wind turbines are subject to variability in energy production due to the intermittent and irregular nature of wind.

The most common design today is the Horizontal-Axis Wind Turbine (HAWT), characterised by a rotor axis parallel to the ground. These typically feature two or three blades and are equipped with an upwind rotor configuration. The selection of specific design features depends on various factors, including aerodynamic efficiency, mechanical complexity, acoustic emissions, and cost.
Modern turbines can be further classified according to two main criteria:

- Vertical axis turbines
- Horizontal axis turbines
- Number of rotor blades

HAWTs are the most widely adopted configuration. They must be oriented to face wind direction, in contrasts to VAWTs, which operate with airflow impacting the blades tangentially. VAWTs, used more commonly in the previous century, resemble the operating principle of classic water wheels. Although, their design tends to be more

complex, they offer the advantage of placing the gearbox and generator at ground level, simplifying maintenance. However, they are generally less efficient due to their lower height, where wind speeds are reduced by the boundary layer effect. In contrast, HAWTs position the rotor at the top of a vertical tower, where higher wind speeds are available, even though this comes at the cost of greater structural load and potentially stronger vibrations due to the increased mass. HAWTs can be further divided based on the rotor's position relative to the wind:

- Upwind rotor
- Downwind rotor

In upwind turbines, the rotor is placed in front of the tower, facing the wind. This configuration helps to avoid the wind shadow created by the tower but requires a yaw control system to maintain alignment with wind direction. Conversely, downwind turbines position the rotor on the side opposite to the wind. While this configuration can eliminate the need for an active yaw system, since the nacelle can rotate passively with the wind, it is more prone to fatigue loads due to flow disturbances and dynamic interactions with the tower wake. Additionally, gyroscopic loads may arise from passive yawing. Upwind rotors typically require stiffer blades to prevent tower strikes, whereas downwind rotors can afford more flexibility, which may reduce weight and lower structural loads. Despite these potential advantages, the vast majority of modern wind turbines employ upwind rotor configurations.

Figure 2.1 :Horizontal axis wind turbines (left) and vertical axis wind turbine (right)

2.2 Main Components of Wind Turbines

The basic structure of wind turbines is typically divided into five main groups: the rotor, the drivetrain, the main frame, the yaw system, and the tower (Fig. 2.2).

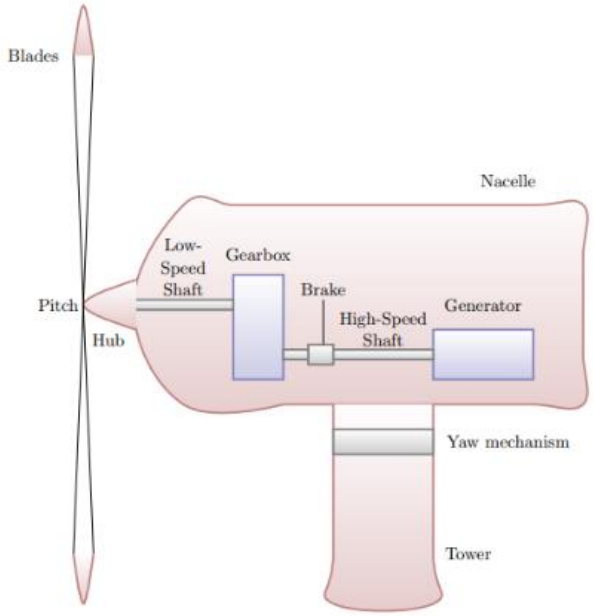

Figure 2.2: Horizontal axis wind turbine representation, and its main component groups

The rotor is the rotating component that extracts wind power and converts it into rotational motion. It is consists of the blades, the hub, and aerodynamic control mechanisms.

- *Blades* are devices that convert lift force into torque, which is transmitted to the rotor. They must be designed considering both their structural strength and aerodynamic performance. In terms of structural strength, they are usually made from composite materials such as fiberglass in polyester resin or carbon fibers. For vertical-axis turbines, aluminium is a common choice. Furthermore, the blades need to be robust enough to endure intense loads that cause significant deflections and fatigue phenomena.
- The *hub* is the part that connects the blades to the main shaft and other drivetrain components. It transmits the torque generated by the blades to the low-speed shaft and is often where the blade-pitch actuator is located. In horizontal-axis turbines, three main types of hub design can be distinguished: rigid hubs, teetering hubs, and hinged blade hubs.

The drivetrain system includes all rotating components of the machine (Fig. 2.4), excluding those described in the rotor section.

- The *main shaft*, also known as the low-speed or rotor shaft, is the primary rotating component that transmits torque from the rotor system to the drivetrain. Additionally, it supports the rotor's weight and must be able to withstand bending and torque loads. Bearings play a crucial role in supporting the main shaft, transferring all reaction loads to the turbine's main frame.
- *Gearboxes* are used to increase the rotational speed of the input shaft to the generator. Wind turbines are large machines that, due to their centrifugal resistance, must rotate at low speeds, while the generator system convert mechanical energy into electrical energy, which is linked to the grid frequency, typically fixed to 50 Hz.

  Gearboxes include torque-transferring components like gears, shafts, bearing, seals, and cases. Generally, gearboxes can be classified into two main types: parallel-shaft gearboxes and planetary gearboxes (Fig. 2.3). Wind turbines operating in industrial settings typically use a three-stage gearbox.

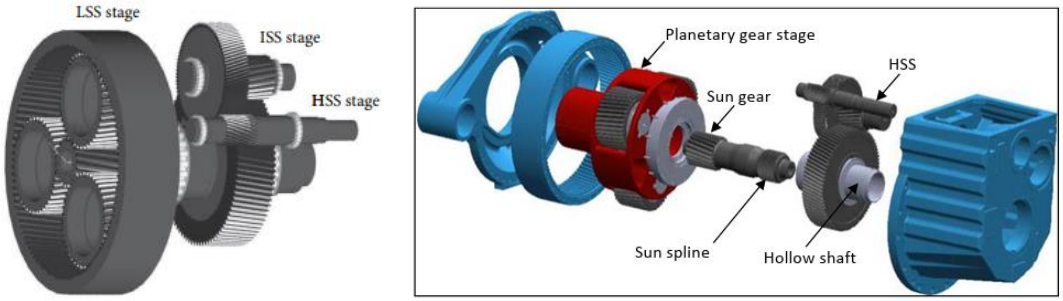

Figure 2.3: Planetary and two stages and gearbox configurations

- The *generator* ensures the conversion of mechanical energy into electrical energy. A converter is used to connect the generator to the AC grid in variable-speed wind turbines.
- *Brakes* are devices employed to stop the rotor or control power output when wind speed exceeds a certain threshold. As such, they are also used during emergency shutdowns.

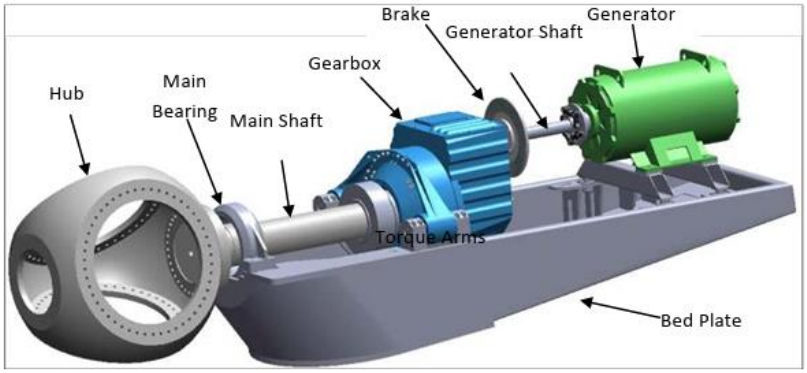

Figure 2.4: Drivetrain representation

The *nacelle* is the enclosure housing the main components of the wind turbine, including the main frame and the nacelle cover. In particular, the main frame is a structural component that supports and connects the gearbox, generator, and brakes. It is a rigid structure designed to transmit both operational and reaction loads from the rotor and generator to the tower. The nacelle cover, on the other hand, provides protection for the turbine's components against weather conditions.

The *tower* is designed to support the entire wind turbine structure, including the nacelle and the rotor. It is typically made of steel, and its diameter is related to the size of the rotor, particularly the length of the blades. As a result, the nacelle is suspended in the air by the tower, which also rotates during power control through the yaw mechanism. However, the yaw system's operation depends on the turbine’s size, as excessive rotation could be dangerous. The yaw mechanism ensures that the rotor aligns with the wind direction, especially during full power production, and can also be used to reduce power output when wind speeds are too high.
The tower is subjected to both static and dynamic loads. Static loads are primarily result from aerodynamic forces that generate torque, while dynamic loads can cause significant vibrational phenomena, particularly in soft towers. In general, the tower's design must ensure that its first natural frequency does not coincide with the turbine’s excitation frequency, which is related to the rotor frequency or the blade passing frequency. A simple method for estimating the tower's natural frequency is to approximate it as a cantilever beam with a point mass at the top, as proposed by Baumeister in 1978.

2.3 Failure Mechanisms

Diagnosing failures in complex machines is particularly challenging, especially when multiple interconnected components are involved. In the case of wind turbines, it is essential to accurately identify the affected subsystem and understand the nature of the underlying faults. A thorough analysis of the most common failure mechanisms not only enhances the interpretation of diagnostic data but also improves the effectiveness of machine learning algorithms used for fault detection and predictive maintenance.
This section provides an overview of the typical failure modes in wind turbines, forming the foundation for the subsequent application of advanced diagnostic and analytical methodologies.
The most common failure in wind turbines involve the *blades*, *gearbox*, *pitch system*, and *yaw system*, as schematically illustrated in Fig 2.5.

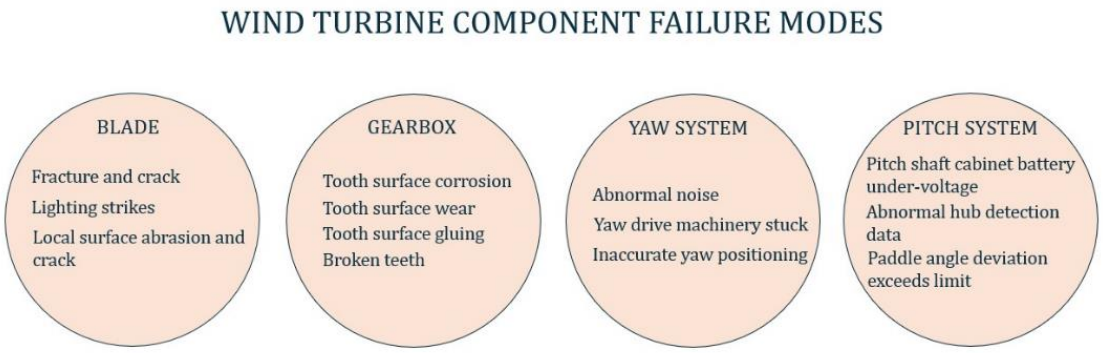

Figure 2.5: Main failure mechanisms in wind turbine

2.3.1 Blade Failure

Wind turbine blades are constantly exposed to the external environment, making them susceptible to a wide range of environmental stressor. These include high altitudes, ultraviolet radiation, heavy rain, freezing rain or ice, snow, lightning, typhoons, as well as sand and dust. In addition, mechanical stress resulting from rapid rotation can amplify these effects and significantly increase the risk of failure.

Blade failures can also stem from material defects such as delamination, folding, or inadequate curing during the manufacturing process. These flaws compromise the structural integrity of the composite materials typically used in blade construction.

As shown in Fig. 2.5, the most common blade failures can be classified into three main categories: fracture and cracking failure, lightning strike damage, and local surface abrasion.

Fracture failures (Fig. 2.7a) occur when a blade breaks into two or more parts. From a material science perspective, structural composites inherently contain micro-defects introduced during manufacturing. Wind turbine blades are often tapered to improve aerodynamic efficiency and reduce weight. However, this design results in a smaller cross-sectional area in the mid-span region, which can compromise stiffness and strength. Under critical external loads, the blade material initially undergoes elastic deformation, storing strain energy. As loading continues, damage mechanisms such as crazing may occur, ultimately leading to fracture.

Another contributing factor is turbulence generated between two successive blades. If the spacing between blades is too wide, airflow disturbance can reduce energy capture and compromise stability. Conversely, if the spacing is too small, blade interference and

excessive turbulence may result, increasing vibration phenomenon and accelerating fatigue failure.

As discussed in the previous section, towers are designed to elevate turbines to altitudes that maximise wind energy capture while reducing turbulence. The site selection for a wind farm is thus critical, and particular attention is paid to the boundary layer in the atmospheric profile (Fig. 2.6).

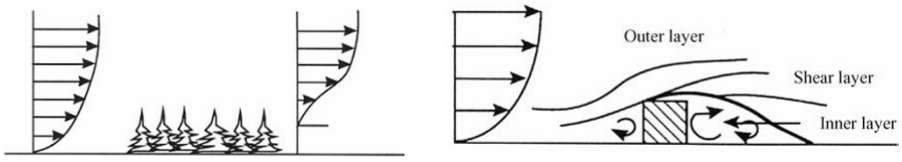

Figure 2.6: Schematic representation of boundary-layer flow over a region of surface roughness and its separation

The figure shows how wind speed varies with both geographic location and altitude above ground level. High-capacity wind turbines often employ taller towers to access more stable and higher wind speeds. However, greater height increases exposure to *lightning strikes* (Fig. 2.7b), which frequently impact the lightning receptor and nearby blade sections. These events may cause cracks in both the lightning protection system and the blade structure itself, often accompanied by visible surface deterioration.

Local surface abrasions (Fig. 2.7c) is also a common form of damage and typically affects the tip and mid-windward sides, as well as the leading and trailing edges of the blade. As the blades rotate, they collide with airborne particles such as sand or water droplets. Due to the increasing tangential velocity along the blade radius, the leading-edge experiences significant erosion, while the trailing edge is more prone to vortex-induced wear. Abrasion from gravel may initiate small depressions that evolve into pitting. Water accumulation in these pits can raise local humidity, increasing the blade's vulnerability to lightning damage.

Figure 2.7: Real images of wind turbine blades affected by different types of failure: structural fracture, lightning damage, and surface abrasion

Wind turbines are equipped with a set of uniformly spaced blades mounted on the rotor, and their interaction with other structural components induces a periodic excitation of the nacelle. Under healthy operating conditions, these interactions are minimal, as the airflow from the rotor is appropriately guided to match the angle of any guide vanes. However, when a blade is damaged or surface abrasions alter the aerodynamic profile, the angle of the fluid flow may deviate from the design conditions. This misalignment leads to more impulsive and unbalanced interactions, which manifest as distinct changes in the vibration signals. Therefore, blade-related anomalies can be effectively detected through vibration monitoring, by analysing variations in vibration amplitude or its frequency content of nacelle vibrations.

2.3.2 Gearbox Failure

Gearbox malfunctions in wind turbines are commonly attributed to gear-related failures, primarily involving the gear teeth. These include corrosion, surface wear, bonding issues, and breakage.

The most typical mechanisms are briefly described below, starting with tooth surface corrosion.

As illustrated in Fig. 2.8, corrosion on the tooth surface may results from either chemical or electrical processes. Chemical corrosion is the most frequent form and is usually caused by an excessive moisture in the lubricating oil or high humidity inside the gearbox casing. Contact pressures exceeding an acceptable threshold, or defects in lubrication that impair surface interaction, can initiate corrosion. When direct metal-to-metal contact occurs under significant compressive loads, it may lead to a substantial energy loss, localised overheating, and, in severe cases, micro-welding of surfaces. This phenomenon is known as seizing.

Figure 2.8: Wear corrosion induced by chemical reactions in the lubricating oil

Surface wear phenomenon can manifest in several forms, including:

- Micro-pitting
- Macro-pitting
- Spalling
- Fretting wear

These forms depend on the intensity and origin of compressive stresses. *Micro-pitting* (Fig. 2.9) also referred to as 'grey spot', occurs on gear teeth subjected to prolonged heavy loads. It is typically the result of incomplete or irregular lubricant film formation, which increases friction and promotes localised heating. These conditions accelerate the initiation of micro-pits on the tooth surface.

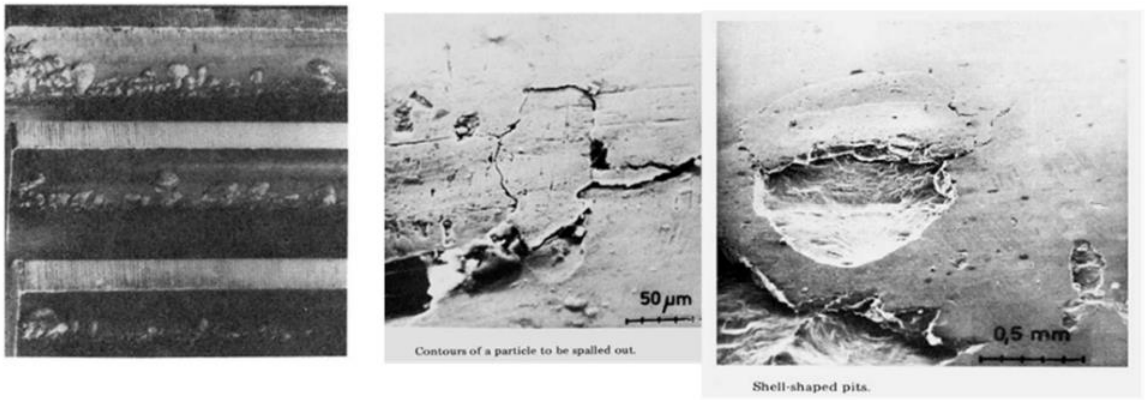

Figure 2.9: Micro-pitting on the gear tooth surface

In contrast, *macro-pitting* (Fig. 2.10) involves the development of larger cavities at gear contact points due to material fatigue. When material defects are present or gear teeth experience strong impacts, high shear stresses can occur from relative sliding. This leads to crack nucleation and propagation along the surface of the gear teeth. As gears continue to operate, these fatigue cracks may extend, resulting in surface spalling or breakage.

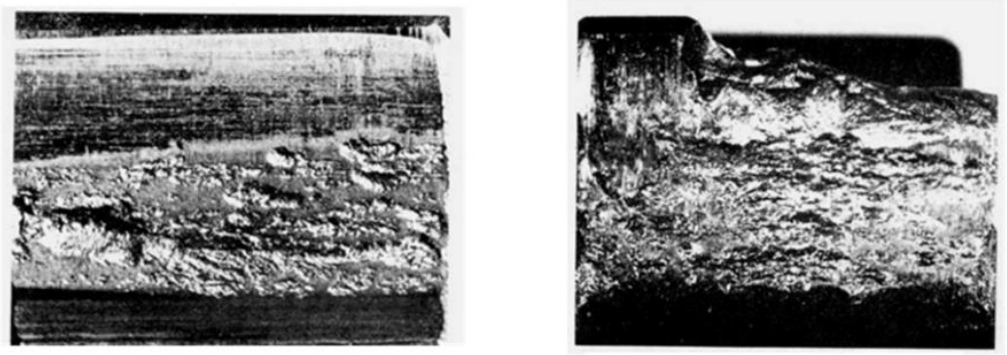

Figure 2.10: Macro-pitting of tooth surface

Fretting wear, or 'black line' corrosion (Fig. 2.11), often arises from intense metal-to-metal contact between gear surfaces during extended periods of inactivity, especially when lubrication is lacking. Small vibrations and insufficient oil flow contribute to the progressive degradation of the surface, which may evolve into pitting and eventual cracking under thermal and mechanical stress.

Figure 2.11: Severe fretting corrosion on a gear tooth

Another degradation mode is *tooth surface gluing***,** which occurs when oil film rupture under excessive pressure allows direct metal contact. This failure mode is categorised into two types:

- Hot gluing, associated with high sliding speeds and temperatures, causes oil film collapse and thermoplastic deformation of gear surfaces.
- Cold gluing, caused by high local pressures under low sliding conditions, leads to oil film puncture and adhesive failure between surfaces.

The final failure mode involves *tooth breakage* (Fig. 2.12), which can result from material fatigue or sudden overloads. This typically occurs when the load surpasses the allowable threshold defined by the material properties. Fatigue fractures exhibit three distinct zones:

- Fatigue crack source zone
- Crack extension zone
- Transient fracture zone

A plastic deformation region is often observed, the extent of which depends on the material's ductility, stiffness, and operating temperature.

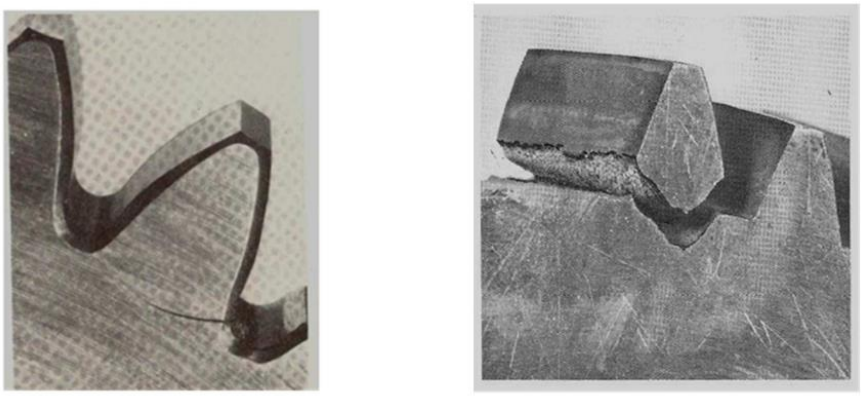

Figure 2.12: Fractured gear tooth due to overload or fatigue

2.3.3 Pitch System Failure

Pitch system failures are relatively common in wind turbines and are typically caused by one or more of the following issues:

- *Undervoltage in the pitch shaft cabinet battery*: This occurs due to battery degradation from prolonged use or failure in the charging circuit. In either case, the battery voltage drops below the critical threshold, preventing the blades from adjusting their pitch angle correctly.
- *Abnormal hub sensor data*: malfunctions in the data acquisition system from the hub may lead to incorrect or delayed pitch adjustments, affecting aerodynamic performance.
- *Exceeding pitch angle limits*: Operational overshoots or improper shutdown sequences may cause the blade pitch angle to exceed its upper or lower limits, potentially resulting in mechanical stress or aerodynamic instability.

When a pitch system fault occurs, the wind turbine may no longer regulate the blade orientation effectively, leading to suboptimal energy production or even total power loss. Additionally, blade misalignment can lead to increased aerodynamic turbulence, increasing the likelihood of abnormal vibration patterns. These effects are typically observable through nacelle or rotor vibration analysis.

2.3.4 Yaw System Failure

The yaw system plays a crucial role in wind turbines, allowing the nacelle to rotate around the tower's vertical axis, thereby ensuring alignment with the wind direction. This

alignment is essential for optimising wind energy capture. However, yaw system failures are relatively common, primarily due to the mechanical components being subjected to continuous stresses, wear, and operational faults.
Typical failure modes can be classified as follows:

- *Abnormal noise*: This issue is usually associated to mechanical components such as the drive motor, gearbox, yaw ring gear, gear engagement, clearance, bearings, and brakes. Failures in these parts often result from lubrication problems. The presence of excessive impurities or lubricant degradation can alter the oil's dynamic viscosity, increase friction and cause localised overheating. Since material mechanical properties are temperature-dependent, elevated temperature can reduce stiffness, making components more susceptible to corrosion and wear. Yaw brake failure is typically identified by the presence of cracks in the friction discs, incorrect adjustment of the calliper body compensation system, or uneven damping torque values. These problems are often caused by excessive braking pressure or contamination of the friction surfaces by dust or debris.
- *Yaw drive machinery blockage*: The yaw drive mechanism may seize due to several contributing factors. First, bearing fatigue may result from thermal expansion and contraction caused by fluctuating operating temperatures. Because the yaw drive is mounted vertically, the lower bearing is subjected t high inertial forces from the yaw ring gear. Foreign particles and debris can infiltrate the bearing gaps, reducing their functionality. Additionally, the yaw gear is highly sensitive to over-torque damage during turbine startup and shutdown. Typical examples of such damage include broken teeth in the yaw ring gear and in the yaw drive gear, as shown in Fig. 2.13a and Fig. 2.13b, respectively.
  Another common failure involves cracking of the yaw bearing liner, often caused by substandard liner materials or improper balancing of the yaw calliper torque.
- *Inaccurate yaw positioning*: Misalignment in yaw positioning can occur due to yaw motor failure, insufficient braking pressure, or inadequate braking torque.

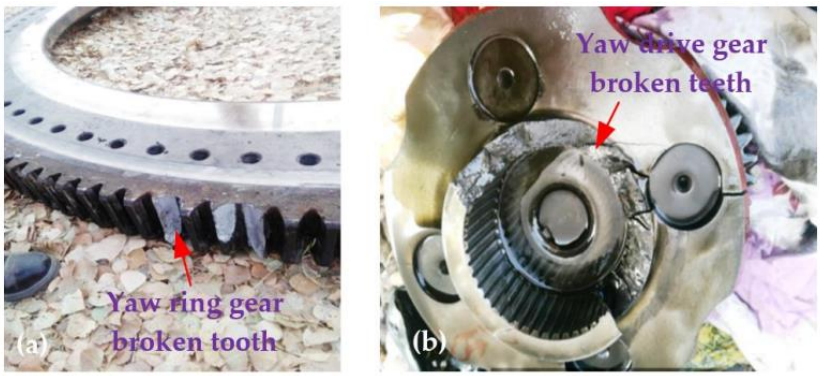

Figure 2.13: Two common failure modes in wind turbine yaw gears: (a) a broken tooth in the yaw ring gear; (b) multiple broken in the yaw drive gear

2.4 Off-design Power Regulation Methods

As mentioned in the previous section, wind turbines are subjected to strong dynamic loads due to their considerable size, requiring constant monitoring and control of the power they generate. Without control mechanisms, the rotor speed would increase as the kinetic energy of the wind increases. Additionally, the centrifugal forces acting on the machine must be considered, highlighting the importance of effective power regulation.
Wind turbines typically operate within specific wind speed ranges, defined by minimum and maximum thresholds:

- Cut-in velocity: $c_{0,min} = 2 \div 3\ m/s$
- Cut-off velocity: $c_{0,max} = 20 \div 25\ m/s$

When wind speed exceeds the cut-off velocity, a portion of the energy must be dissipated to avoid mechanical damage. Conversely, if the wind speed is below the cut-in threshold, the energy extracted is insufficient to overcome the internal resistances of the turbine components, leading to inefficiencies.
Moreover, the number of blades plays a crucial role in turbine performance, as it influences turbulence generation, which can lead to vibrational phenomena in the nacelle. This aspect is quantified by the solidity, defined as:

$$s = \frac{Total\ Blade\ Surface}{Disk\ Actuator\ Area} \tag{2.1}$$

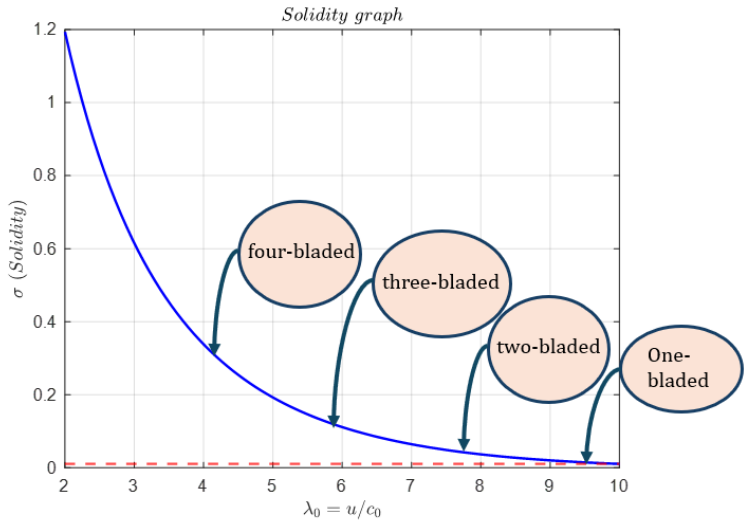

Figure 2.14: Solidity graph

As shown in Fig. 2.14, turbines with a single blade are typically used for small-scale energy generation in local facilities or energy communities, often driven by sustainability

goals. For turbines with low solidity, only a small fraction of the air stream interacts with the rotor. As a result, the rotor must spin faster to induce sufficient perturbation of the airflow, allowing maximum energy extraction.

There are three main methods used to control power output in off-design conditions:

- Pitch control
- Stall control
- Active stall control

Pitch control modifies the aerodynamic performance by rotating the blades around their longitudinal axis. Specifically, the electrical power output is influenced by the tip-speed ratio λ_0, which is affected by changes in the tangential velocity. When the generated power exceeds the turbine's maximum allowable limit, typically defined by centrifugal constraints, the blades are slightly pitched out of the wind to reduce power. This method requires a blade design capable of maintaining the optimal angle across a range of wind speeds. Pitch control is often combined with rotor speed regulation to enhance efficiency.

Stall control reduces power output by exploiting aerodynamic stall. Blade profiles with high Reynold numbers tend to stall more easily. As wind speed, and consequently rotor speed, increases, the stalled region on the blade surface expands, reducing aerodynamic lift. To facilitate this, the blade geometry is designed to induce flow separation on the leeward side once the wind exceeds a critical velocity.

Active stall control is a hybrid approach that combines the previous two methods. At low wind speeds, the turbine operates similarly to a pitch-controlled system. At higher speeds, however, the blades are deliberately pitched to induce stall and reduce power output, thus preventing overspeed and mechanical stress.

2.5 Thesis Purposes

The primary objective of this research is to explore and develop effective anomaly detection strategies for wind turbines by applying machine learning algorithms applied to SCADA data. Specifically, the aim is to accurately predict nacelle vibrations, which serve as continuous monitoring indicators of the turbine's normal operating conditions. By establishing precise thresholds, the system is designed to identify anomalies corresponding to significant deviations from expected behaviour.

Anomaly detection is a crucial approach for improving predictive maintenance procedures as highlighted on the introduction of this thesis. Although several challenges must be addressed before initiating a reliable analysis. The literature is rich in journal articles and technical reports that propose a variety of algorithms based on machine learning techniques or traditional statistical models, all aimed at overcoming the inherent difficulties of this field. The continuous advancement of information and electronic technologies offers many advantages, from improved data acquisition and sampling to the development of increasingly accurate models. Among the various research contributions in the field, the work presented in [31] is taken as the primary reference for developing the methodology proposed in this thesis.

To fulfil the overarching aim, a key goal is the integration of physical modelling with data-driven approaches to predict nacelle vibrations, which are used as indicators of potential faults in the turbine and form the basis of vibration monitoring systems. More specifically, this work aims to develop a regression model that links operational parameters, such as wind speed, rotor speed, pitch angle, and others, to nacelle vibrations. These vibrations are critical indicators of the mechanical health of the turbine and accurately predicting them can help detect deviations from normal behaviour, indicating possible malfunctions.

To this end, a comprehensive approach is adopted. First, the relationship between operational parameters and nacelle vibrations is examined through a preliminary statistical analysis using a two-input, single-output regression model. Wind turbines operate under complex and dynamic conditions and understanding the interplay between these parameters is essential to accurately model the behaviour of the nacelle across various operational scenarios.

In addition to the regression model, the methodology integrates physical insights based on the Betz model, which describes how wind turbines extract kinetic energy from the wind by reducing its velocity and increasing the flow area. This integration of physical knowledge into the machine learning process enhances both its accuracy and robustness, making it more suitable for real-world applications.

Furthermore, the proposed methodology includes a detailed residual analysis. After training the regression model on data representing normal operating conditions, the residuals, defined as the differences between predicted and actual vibration values, are examined to identify anomalies. By defining specific thresholds for the residuals, it becomes possible to detect potential faults or abnormal behaviour in the turbine's operation.

The dataset used in this thesis consists of SCADA data collected from multiple wind turbines over several months. This dataset, available from Zenodo under the tile '*Wind Turbine SCADA Data for Early Fault Detection*', offers a rich and high-dimensional source of information, ideal for developing a robust predictive model.

The aim of this work is to improve the accuracy of predictive maintenance systems for wind turbines, reduce operational costs, and enhance the overall reliability of turbine operations. Through a combination of machine learning and physical modelling, this research contributes to the advancement of sustainable and efficient wind energy technologies, with a focus on early anomaly detection.

3. Dataset Description and Initial Preprocessing

To provide a clear overview of the entire work, this chapter presents a detailed description of the dataset used for the analysis, including the selected wind turbine and the features considered. Furthermore, a brief overview of the typical measurement instruments employed during the data acquisition process is provided. The chapter concludes with a description of the data cleaning methodology adopted prior to the modelling phase.

3.1 Dataset

The dataset used in this thesis was downloaded from Zenodo under the title '*Wind Turbine SCADA Data For Early Fault Detection',* uploaded by Gück et al. [36]. According to the Zenodo platform, this dataset was made available recently, and to date, it has not yet been widely used in the scientific literature. Therefore, it is expected that this work may provide a valuable contribution to the research fields of anomaly detection and condition monitoring.

The corresponding publication introduces high-dimensional data collected from three different wind farms located in Portugal and Germany. The complete dataset comprises 95 individual time series, representing 89 years of SCADA data across 36 wind turbines. Furthermore, it includes detailed information about the operational status of the turbines, indicating whether they were functioning normally or exhibiting abnormal behaviour. These labels are extensively used in this work for performance evaluation of the anomaly detection system, using metrics such as accuracy, missed alarms, and false alarms. Each labelled event is annotated with a start and end timestamp, along with a textual fault description.

In the field of anomaly detection, data quality and the level of detail are crucial. This is not only reflected in the presence of reliable labels but also in the availability of documentation that identifies the root cause of each anomaly and whether it is linked to a physical failure. The effectiveness of any analysis is strongly dependent on the quality and acquisition process of the input data. In wind turbine condition monitoring, SCADA data (Supervisory Control and Data Acquisition) are widely used. These systems are capable of acquiring extensive data through a network of dedicated sensors, which enhances both the monitoring and the autonomous diagnostic functions of wind turbines. It continuously logs a wide range of parameters, including environmental, mechanical, thermal, and hydraulic data, even under changing ambient conditions. In real-world

application, such as in the dataset used for this thesis, SCADA is widely employed to identify potential faults at an early stage. This early detection capability plays a key role in enabling proactive maintenance strategies and ensuring efficient operation of wind farms, with the added benefit of reducing overall maintenance costs.

Wind farms are typically classified as onshore, offshore, or nearshore, depending on their location relative to land or sea (Fig. 3.1). specifically, the three wind farms in this dataset are anonymised and referred to as A, B, and C:

- A: Onshore wind farm in Portugal with 5 wind turbines
- B: Offshore wind farm in Germany with 9 wind turbines
- C: Offshore wind farm in Germany with 22 wind turbines

Figure 3.1: Example images of offshore, onshore, and nearshore wind farms

Wind farm A is based on the EDP-data, sourced from the EDP-platform. This is one of the most relevant public datasets in the field of early fault detection, as it includes not only SCADA-data, but also fault information. However, despite its popularity, the fault-related information it provides is relatively limited. Wind farm A includes 22 datasets, whereas wind farms B and C together include 73 datasets.

In this work, the analysis focuses on wind farm C, as discussed in Chapter 1. The main objective of this thesis is to develop a method for detecting damage-related anomalies based on vibration monitoring. Among the three wind farms, only wind farm C includes vibration measurements, making it the only suitable source for the analysis. It provides 58 datasets in total. Each dataset is provided as a .csv file, where columns represent features and rows represent time series data points.

Each time series includes five key descriptive columns: a row ID, a timestamp, an asset ID identifying the turbine, and a status-ID indicating the turbine's operational condition at each time point. Table 3.1 presents an overview of the three wind farms, including the number of turbines, datasets, and whether these datasets represent normal or abnormal behaviour.

Table 3.1: Wind farms overview

	Wind Farm A	**Wind Farm B**	**Wind Farm C**
Turbines	5	9	22
Datasets	22	15	58
Anomaly Events	11	6	27
Normal Behaviour	11	9	31

Regarding labels, the data is annotated on two levels. The first level concerns the so-called *event labels*, which are assigned at the dataset level. If a dataset contains an anomaly event within the defined prediction timeframe, it is labelled as anomalous; otherwise, it is labelled as normal. These anomaly labels have been established either through direct feedback from wind farm operators or through documented faults found in service reports and fault logbooks. Conversely, the labels identifying normal behaviour have been defined based on a of operator feedback, manual inspections of the data, and expert knowledge.

In the specific case of wind farm A, the start of each anomaly event was determined using the EDP fault logbook, which provides the fault timestamps but not additional contextual information. For wind farms B and C, the identification of anomaly event starts was more detailed and was carried out through a combination of data analysis, operator feedback, service report documentation, and expert evaluation. Although there is a possibility that the actual onset of certain anomalies might differ slightly from the annotated ones, it is considered highly unlikely that the recorded events begin prematurely. If anything, the annotated start-times are more likely to be later than the actual fault onset.

The second level of labelling is performed at the timestamp level and involves the assignment of a *status-ID* to each data point in the time series. For wind farms B and C, these labels were derived from the original operating modes provided by the wind farm operators, combined with information extracted from service reports. For wind farm A, however, such detailed operational data were not available, and therefore the status-ID values were assigned based on EDP fault logbook. In this context, for each fault recorded, the 14 days prior to the fault were labelled with status-ID 4, indicating a fault condition (Table 3.2), while the 3 days following the fault timestamp were labelled with status-ID 3, corresponding to the service mode.

Table 3.2: Description of status-ID labels

Status	Description
0	Normal operation without limitations
1	Derated power generation (with power restriction)
2	Turbine is idling and waiting to operate again
3	Turbine is in service mode (service team on the site)
4	Turbine is down due to a fault or other reasons
5	Other operational states (e.g., system test, setup, ice build-up, emergency power)

These status labels are particularly useful for determining whether a specific data point reflects normal or abnormal turbine behaviour.

3.2 Data Acquisition

Condition monitoring systems consist of a combination of sensors and signal processing units that continuously provide information about the condition of components through techniques such as vibration analysis, acoustics, oil analysis, and performance or thermal monitoring. The data acquisition process is generally not straightforward, as it usually involves multiple stages, as illustrated in Fig. 3.2.

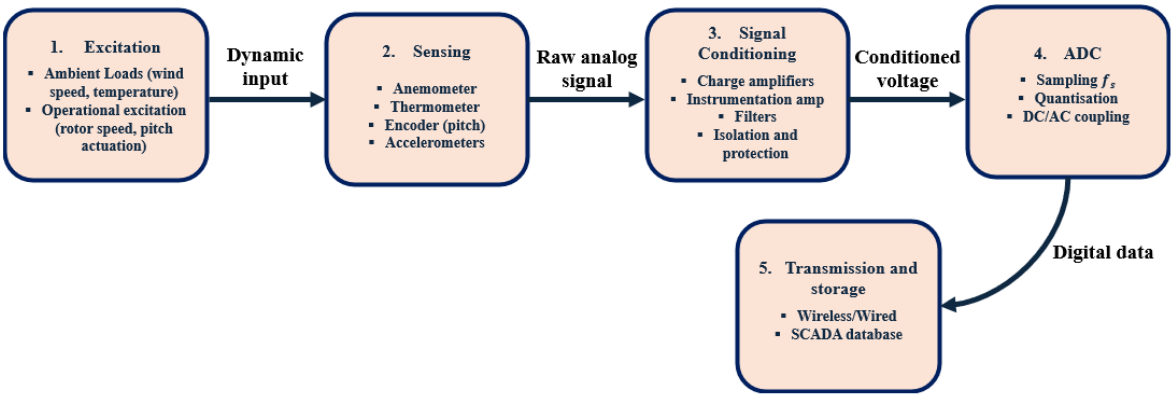

Figure 3.2: Block diagram of the overall data acquisition workflow

3.2.1 Generation of Dynamic Input

The first phase of data acquisition is *excitation*, which initiates the generation of a dynamic response. A dynamic input is required to induce this response, specifically, a load with a frequency excitation higher than the first natural frequency of the structure. Typically, in controlled experiments, such as applying a shaker to a beam, the input can be managed to investigate vibration modes across different frequencies. When the excitation is both controlled and recorded, the output can be normalised to the input and analysed in the frequency domain as a Frequency Response Function (FRF), a system-invariant measure independent of excitation amplitude.

In wind turbine applications, the excitation is largely dependent on environmental conditions, such as wind speed, temperature, humidity, and others, or internal system dynamics. In these cases, the input cannot be directly controlled, and it is not possible to determine a priori which phenomena contribute to vibration. Nevertheless, in this work, several physical parameters are selected as input variables to build the input matrix for training the SVR model:

- *Wind speed*
- *Ambient temperature*
- *Pitch angle*
- *Position rotor blade axis*
- *Gearbox oil temperature*
- *Rotor speed gearbox main shaft*
- *Axis rotor bearing inner ring temperature*

Each of these physical parameters can be measured by variety dedicated sensors. These sensors are often evaluated based on criteria such as cost, size, mass, reliability, stability, and dynamic performance (e.g., frequency resolution, bandwidth, sensitivity). Sensors can be classified as *contact sensors* (e.g., piezoelectric, piezoresistive, capacitive), or *non-contact sensors* (e.g., acoustic emission, strain gauges, accelerometers). Additionally, sensors can be categorised by the number of physical quantities that can measure, as some are multi-functional.
The following subsection shows an overview of the sensors used for each input parameter in this study.

3.2.2 Sensor Instrumentation and Conditioning

Wind speed can be measured using sonic anemometers, which detect three-dimensional wind vectors, temperature, and surfaces fluxes. These devices can measure wind speed form 0 to $60\ m/s$ at sampling rates up to $100\ Hz$. A widely used model is the Campbell Scientific CSAT3 shown in Fig. 3.3. Alternatively, wind speed profiles can be obtained by mounting simple anemometers at various tower heights. There are three main types of anemometers, as depicted in Fig. 3.3.

Figure 3.3: Images of cup anemometers, propeller anemometers, and sonic anemometers

All types are low power ($\leq 1\ W$) and suitable for continous monitoring. Guidelines for installing cup anemometers on towers are found in the International Electrotechnical Commission (IEC) 61400-12-1 (2005), which serves as the standard for wind resource assessment and high-quality monitoring. Cup anemometers measure wind speed by detecting the rotational speed of multiple cups mounted on a vertical axis. The rotation, induced by the wind, is converted into an electrical signal through an encoder. Propeller anemometers operate similarly, employing a propeller attached to an aerodynamic body to detect wind-induced rotation. In contrast, sonic anemometers determine wind speed by measuring the time of flight or phase shift of ultrasonic pulses transmitted between pairs of transducers. The wind affects the propagation time of the sound wave, allowing the device to infer the wind speed along a defined path, typically no longer than 20 cm. depending on the configuration, sonic anemometers can perform measurements in one, two, or three spatial dimensions.

Meteorological variables, such as *ambient temperature,* are typically measured near ground level using thermometers, barometers, and hygrometers mounted at standard reference heights of 2 or 10 meters. Although, sonic anemometers can also estimate air temperature, but the readings they provide are influenced by both temperature and humidity, representing not a direct temperature value but rather a derived parameter. Therefore, for accurate measurements of ambient air temperature, dedicated thermometric instruments are preferred.

The *pitch angle* is commonly measured using absolute rotary encoder mounted on the pitch gear. These sensors provide angular measurements with a typical accuracy of $\pm 0.1°$. In wind turbines equipped with hydraulic pitch systems, linear encoders may also be employed to track actuator displacement, offering comparable precision.

The *rotor speed gearbox the main shaft* (including rotor speed) is measurable, can also be measured using absolute rotary encoders or proximity sensors. These sensors are capable of detecting motion over a complete rotational cycle $(0 \div 360°)$, though some may require resetting or re-referencing after each full revolution, depending on the system configuration.

The *inner ring temperature of the rotor bearing* is typically monitored using self-adhesive temperature sensor, affixed to the bearing's internal surface and secured with axial epoxy resin. In contrast, the *gearbox oil temperature* is usually measured at the oil sump, where thermal equilibrium is representative of system-wide lubrication conditions.

The primary output variable of the present study, such as nacelle vibration, can be characterised through different physical quantities: displacement, velocity and acceleration. Among the available transducers, only a subset is practically suitable for continuous monitoring in wind turbine applications:

- Proximity probes (relative displacement): non-contact sensors such as capacitive, inductive, eddy current, magnetic, and Hall-effect devices that measure the distance between the probe tip and the target surface.
- Optical displacement sensors: non-contact systems including laser interferometer, Triangulation sensors, time-of-flight lasers and chromatic confocal devices, all of which infer position by analysing reflected light.
- Velocity transducers: contact sensors of a coil mounted on a seismic mass and suspended in the field of a permanent magnet; the relative motion induces a voltage proportional to the velocity.
- Accelerometers: contact transducers based on piezoelectric crystals housed against a seismic mass; mechanical strain within the crystal generates an electric charge proportional to the applied acceleration.

In vibration monitoring applications, accelerometers are the most widely adopted, due to their high sensitivity, broad frequency response, and robustness. The sensors generate an analogue voltage signal, which is then conditioned and forwarded to the digital acquisition system. For instance, piezoelectric accelerometers produce an electrical charge proportional to the applied acceleration. In most practical applications, this charge is converted into a calibrated voltage using an integrated charge-amplifier circuit employing precision operational amplifiers. Conversely, cup anemometers translate the wind-driven rotation of the cups into an electrical voltage via a potentiometer mounted

on the vane, whose resistance changes with the vane's angular position. In both cases, the raw analogue signal is subsequently conditioned, through amplification, filtering, and, where applicable, frequency-to-voltage or current conversion, before being fed into the analogue-to-digital-converter.
In the present work, the wind farm under analysis is an offshore installation located in Germany, and vibration signals were acquired using accelerometers mounted on the nacelle.

3.2.3 Data Transmission and Digital Acquisition

The acquisition system relies on a continuous data stream from the field sensors to a central data logger or controller. To ensure efficient turbine operation, it is crucial that the acquired data are accurate, timely, and minimally distorted. Two main communication types are commonly employed:

- *Wired transmission*: based on copper conductors or optical fibers, capable of transmitting analogue or digital signals with high reliability and low latency.
- *Wireless transmission*: utilises radio frequency (RF) or optical electromagnetic waves and is especially useful in scenarios where cable installation is impractical.

Among wired technologies, the RS-485 interface is widely adopted in wind turbine automation due to its long-range capability, high noise immunity, and multidrop support. Wired solutions offer real-time diagnostics and greater stability, whereas wireless alternative are preferred for remote or rotating parts where cable routing is problematic.

A commonly used digital acquisition platform in wind turbines is the SCADA system, i.e. Supervisory Control and Data Acquisition. SCADA systems continuously monitor and store hundreds of variables, offering insights into the operational health of the turbine. While the data are usually made available to end-users as 10-minute averages, the actual sampling frequency can reach several hertz, enabling more detailed offline analysis.

3.2.4 Analog-to-Digital Conversion

Digital signals offer several advantages over analogue ones, particularly in terms of storage and robustness. Digital data can be easily archived and accessed without the need for bulky physical media, and it is inherently more resistant to noise. Unlike analogue signals, which are vulnerable to degradation and interference, digital signals stored on

electronic media such as hard drives, solid-state drives, or flash memory remain stable over time. Moreover, digital formats support compression techniques that allow for a significant reduction in data volume without compromising the integrity of the information.

A key step in digital signal processing is the conversion of analogue outputs from sensors into digital format, a process that requires discretisation in both time and amplitude. This conversion depends on two primary parameters: the sampling frequency and the amplitude range.

- *Sampling frequency*: The system captures signal values at uniform time intervals, where each sample corresponds to a time step $\Delta t = 1/f_s$. To avoid aliasing, i.e. a phenomenon where higher frequency components are misrepresented in the sampled signal, an analogue anti-alias filter is typically applied before sampling. This filter restricts the signal's frequency content to a cutoff frequency f_c, generally chosen such that the sampling rate $f_s = 2.56 f_c$. This ensures compliance with the Nyquist criterion, which dictates that $f_{Ny} = f_s/2$. While increasing f_s improves temporal resolution, it also results in larger datasets and higher storage demands.
- *Amplitude range E and Bit Depth B*: During quantisation process, the continuous amplitude of the analogue signal is approximated to the nearest discrete level. The resolution of this process, denoted as ΔA, depends on the amplitude range and the number of bits used by the analogue-to-digital converter (ADC), following the relation $\Delta A = E/(2^B - 1)$. Optimising the range E allows for enhanced resolution while also avoiding saturation caused by amplitudes exceeding the limits. Additionally, the choice between DC and AC coupling must be considered. When the signal's mean value carries no significant information, AC coupling can be used to eliminate it, making more efficient use of the ADC's dynamic range.

Once converted, the digital signal is ready to be stored within the data acquisition system for further analysis for monitoring.

3.3 Features Selection

The initial preprocessing phase typically consists of selection a set of statistical features. The health status of a wind turbine is assessed based on the identification of 'abnormal' variations in some of the selected features, which may be associated with the presence of potential damage or malfunction.

In this case, the datasets corresponding to the three wind farms do not contain raw data, but rather precomputed time features. This is likely because industrial SCADA systems generally store data as statistical features at a fixed sampling resolution, typically, as in this case, every 10 minutes.

For each measured physical parameter, the following features have been extracted: average (mean value), maximum, minimum, and standard deviation. The total number of features for each wind farm is summarised in Table 3.3:

Table 3.3: Total number of features per wind farm

	Wind Farm A	**Wind Farm B**	**Wind Farm C**
Features	86	257	957

In signal analysis, statistical features are widely used to characterised signals from both a probabilistic and a time-domain perspective. Some of these features, such as the average and the standard deviation, are directly related to the signal's probability distribution. Others, like maximum and minimum values, are purely descriptive and reflect the signal's variation range over the observed time window.

To formalise the probabilistic interpretation, let us consider a discrete random variable $x(k)$ associated with the experiment k. The Probability Density Function (PDF) of the variable is defined as:

$$p(x) = \lim_{\Delta x \to 0} \left(\frac{Prob[x < x(k) \leq x + \Delta x}{\Delta x} \right) \tag{3.1}$$

This function describes the probability that a value of $x(k)$ is within the interval $(x, x + \Delta x]$. Among the most commonly used PDF is the Gaussian (or normal) distribution, which is also considered in this work. It is defined as:

$$p(x) = \frac{1}{\sigma_x \sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu_x}{\sigma_x}\right)^2} \tag{3.2}$$

as illustrated in Fig. 3.5.

When dealing with a random process, i.e. a set of all possible sample functions $\{x(t)\}$ consisting of N distinc realisation $x_k(t)$, the following statistical features can be defined:

- *Average*

  The mean represents the expected value of the signal and provides a measure of its central tendency. It is calculated as.

$$\mu_x = E[x(t)] = \lim_{N\to\infty} \sum_{k=1}^{N} x_k(t) \tag{3.3}$$

- *Standard Deviation*

  The standard deviation quantifies the dispersion of the signal around its mean and is defined as the square root of the variance

$$Var\big(x(t)\big) = E[(x_k(t) - \mu_x)^2] \tag{3.4}$$

$$\sigma_x = \sqrt{Var(x(t))}$$

  When the mean is zero, the standard deviation equals the Root Mean Square value (RMS). In general, their relationship is:

$$\sigma_x^2 = RMS^2 - \mu_x^2 \tag{3.5}$$

  For stationary zero-mean process, the standard deviation corresponds to the signal's average power.

While the maximum and minimum values are not statistical moments but rather descriptive indicators of the signal's amplitude range over a specific time interval:

- *Maximum*

$$\max(x(t)) = \max_{1\le k\le N} x_k(t) \tag{3.6}$$

- *Minimum*

$$\min(x(t)) = \min_{1\le k\le N} x_k(t) \tag{3.7}$$

Although not directly linked to the underlying probability distribution, the mean value and standard deviation are widely used to characterise the extremal behaviour of a signal over time. The mean provides a measure of the signal's central tendency, while the standard deviation quantifies its variability. In particular, when standard deviation increases, the values of the signal tend to be more dispersed and are more likely to deviate significantly from the mean. For signals that are approximately symmetric, the mean represents the centre of symmetry, and the spread of the values around it is governed by the standard deviation (Fig. 3.5).

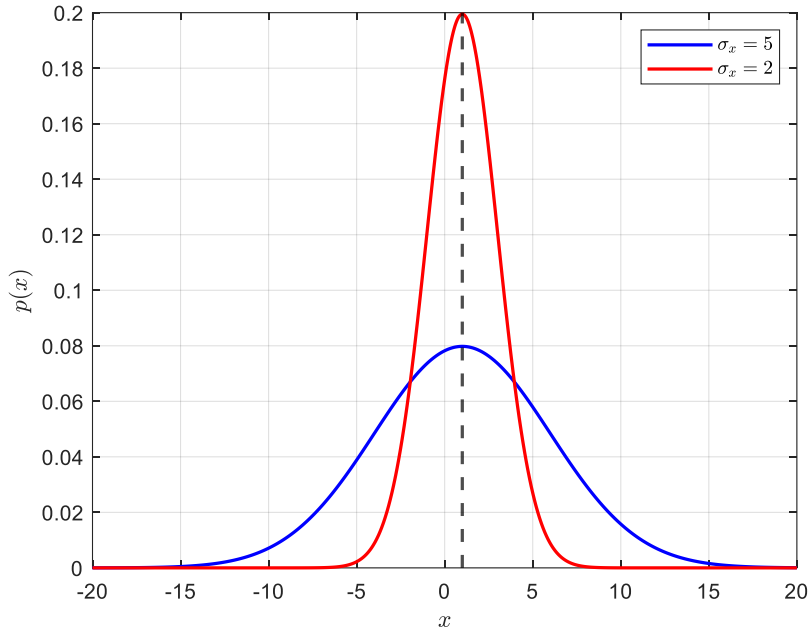

Figure 3.4: Comparison between two normal distributions with different standard deviations

In this thesis, the Gaussian distribution is considered in several stages of the analysis. Its first application occurs during the initial statistical assessment, where a Quantile-Quantile plot (QQplot) of the residuals is used to verify whether their distribution follows a normal one, or exhibits heavy-tailed, non-linearities or asymmetries. The second application concerns the comparison between the distributions associated with healthy data and those corresponding to damaged conditions, in order to evaluate how well the two classes are separated (Fig. 3.6).

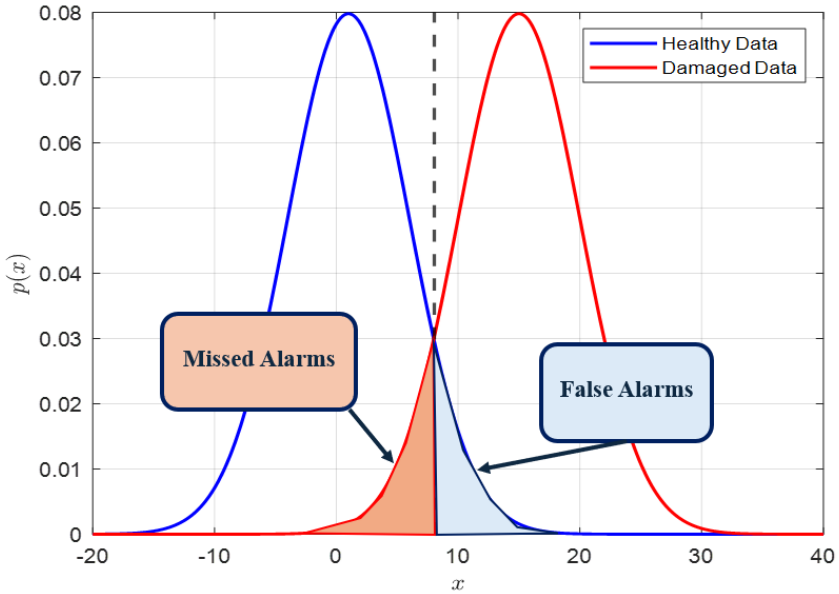

Figure 3.5: Comparison between healthy and damaged distributions

The statistical features selected for this analysis include all four described previously. However, particular attention has been paid to vibration signals, for which only the standard deviation was retained. This choice is motivated by its intrinsic energetic significance: in the case of stationary zero-mean processes, the autocorrelation function at lag zero is given by:

$$R_{xx}(\tau = 0) = \lim_{T \to \infty} \frac{1}{T} \int_{-T/2}^{T/2} x_k^2(t) dt \tag{3.8}$$

This quantity represents the energy of the signal, and coincides to the standard deviation, when the mean is zero. Hence, for vibration data, the standard deviation is not only statistically meaningful but also physically relevant, as it provides a direct measure of the signal's energy content.

In addition, in the context of machine learning, certain issues must be addressed to ensure robust generalisation between the training and testing phases. One critical challenge is the *data shift*, i.e. when the data distribution changes across different wind turbines. When this occurs, the model's predictive capability degrades, leading to an increase in false positive and missed detections, which do not reflect the actual health status of the machine. In the case of vibration data, incorporating the mean value can exacerbate this issue, as illustrated in Fig. 3.7, which shows vibration output predicted by Support Vector Regression (SVR) model, applied to a turbine different from the one used during training phase.

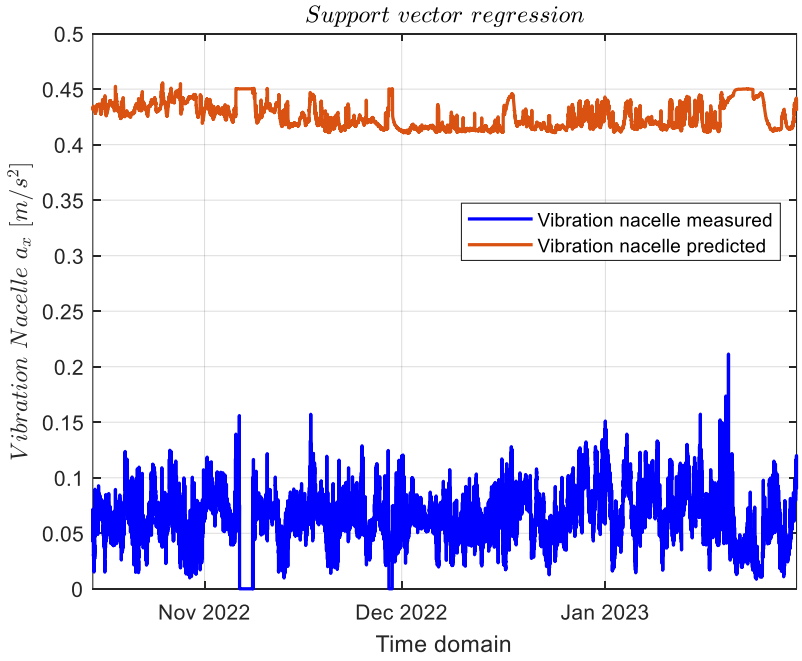

Figure 3.6: Support Vector Regression test on a turbine different from the one used for training phase

3.4 Data Cleaning

After feature selection, data cleaning is the next essential step. Starting from the extracted time-series features of the various physical parameters contained in the dataset, the main objective is to train an SVR model using data from a health turbine. In this context, a commonly technique for identifying potential outliers is based on the median and the median absolute deviation (MAD), which are both robust to non-Gaussian distribution and extreme values. The threshold for outlier detection is defined as:

$$\varepsilon = median\big(x(t)\big) \pm K \cdot MAD\big(x(t)\big) \tag{3.9}$$

This formulation allows for the definition of upper and lower bounds, beyond which any data point can be classified as an outlier. While a typical choice for the constant K is 3, in this work a more conservative value of $K = 4.5$ has been adopted (Fig. 3.8). This adjustment is intended to prevent the loss of potentially relevant information, especially given the non-stationary nature of data recorded from wind turbines.

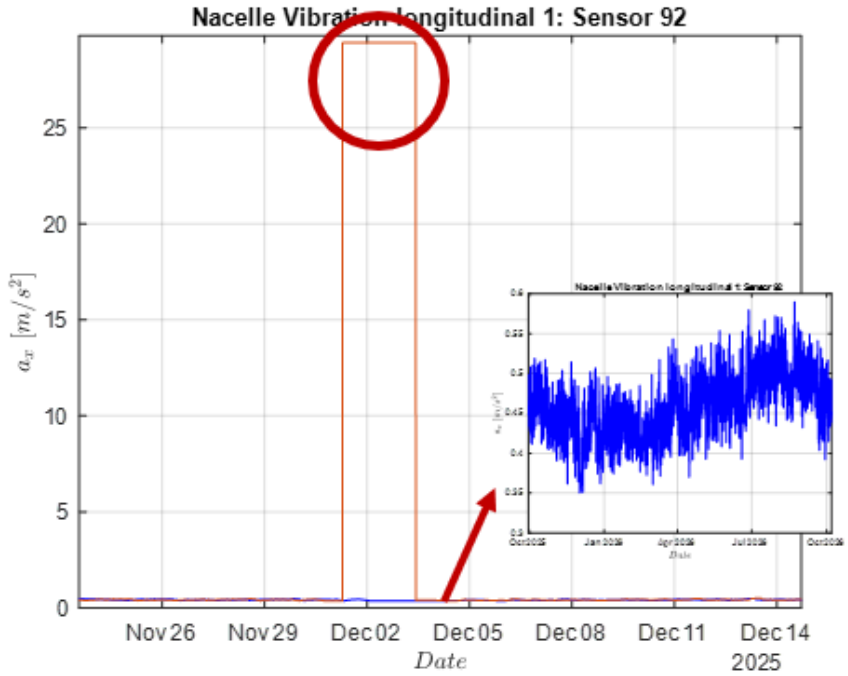

Figure 3.7: Time series of the mean nacelle vibration filtered using MAD-based thresholding. The highlighted anomaly is excluded by the threshold, resulting in the final extracted signal (blue line)

A more detailed discussion of the complete methodology adopted for the development of the anomaly detection system, is presented in chapter 4.

4. Methodology

This chapter outlines the methodology used to develop an anomaly detection model for wind turbines, based on nacelle vibration data. It presents the main steps of the data processing pipeline, the integration of a physics-based wind power estimation through Betz's law, and the use of SVR for modelling. All implementation steps were carried out in MATLAB, combining theoretical models and practical tools to address the detection of pitch related faults.

4.1 Method Overview

As introduced in Chapter 1, condition monitoring methods based on data-driven approaches require a preliminary phase of data preprocessing. This step is crucial to ensure that the input data is adequately prepared for the analysis, as it allows for the reduction of noise and the enhancement of meaningful information within the dataset.
In this thesis, the dataset already includes a set of predefined features, as described in Chapter 3. Therefore, the preprocessing phase is mainly focused on cleaning these features to reduce measurement errors and inconsistencies that might affect the model's accuracy. The general workflow developed for this study is shown in Fig. 4.1, and it consists of several stages before the actual anomaly detection task is carried out.

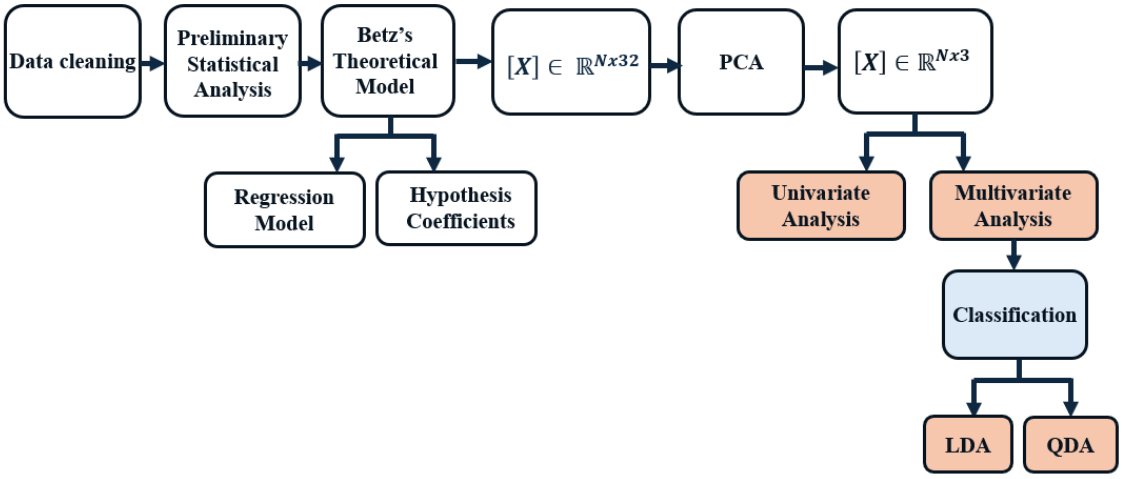

Figure 4.1: General Workflow including preprocessing Steps

The preprocessing phase begins with a data cleaning procedure, which is applied to the available features, in order to minimise the influence of outliers and noise. Once the data

is cleaned, a statistical analysis is performed on a Multiple-Input and Single-Output (MISO) regression model, initially using two input variables and one output. This step is intended to gain insight into the operational behaviour of the wind turbine and to identify which physical parameters are most relevant for vibration signal, measured on the nacelle. At the same time, it helps in reducing the dimensionality of the problem by discarding variables with weak correlations.

In recent years, increasing attention has been given to the integration of physical knowledge into machine learning models, in order to improve their interpretability and generalisation capabilities. In this context, a theoretical model based on Betz's law is introduced. This model provides a first estimate of the power extracted from the wind and transferred to the turbine rotor, accounting for the non linear effects related to wind-turbine interactions.

The approach used here includes two main steps:

- First, a physical formulation of wind power is established, where wind speed is raised to the third power, and the remaining variables in the Betz equation are approximated through reasonable engineering assumptions.
- Then, a regression model is trained using this estimated power as one of the input features, enabling the model to capture the underlying physics while learning from the data.

After these steps, a full input matrix is constructed, initially composed of 32 features. However, since the objective is to train a SVR model, a Principal Component Analysis (PCA) is applied to reduce the dimensionality of the input space and improve computational efficiency. This results in a reduced matrix with 3 principal components, which retains most of the variance of the original dataset.

Once preprocessing is completed, the analysis proceeds through two approaches: a univariate method based on residuals and a multivariate strategy involving multiple accelerometers.

Before training the SVR models, two wind turbines are selected for the analysis, as illustrated in Fig. 4.2, and summarised in Table 4.1.

Table 4.1: Representation of wind turbines considered during analysis

Wind Turbine	Wind Turbine Status	Use
WT53	Healthy	Training
WT53	Healthy	Validation
WT35	Damaged: Pitch-failure	Test

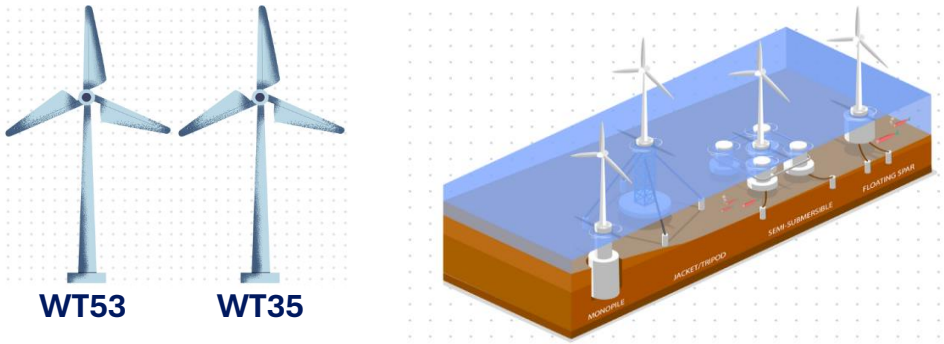

Figure 4.2: Offshore wind farm and selected turbines

Turbine WT53, which operates under healthy conditions, is used both for training and validation, by splitting the dataset into two subsets. The second subset is used for validation purposes, in order to assess the model's ability to predict vibration levels, measured in terms of standard deviation. For the test phase, turbine WT35 is used, which has been affected by a pitch failure. This type of damage is of particular interest because it increases turbulence due to the incorrect blade angle regulation, and as a result, leads to increased structural vibrations, which can be detected by sensors placed inside the nacelle.

The time ranges used for training, validation, and testing are detailed in Fig. 4.3 and Table 4.2.

Figure 4.3: Time segmentation of the dataset

Table 4.2: Time periods for each phase

Time Domain	Use
30/09/2022	Start Training
04/08/2023	End Training
01/09/2023	Start Validation
15/10/2023	End Validation
15/10/2022	Start Test
30/01/2023	End Test

Two separate approaches are then used to analyse the data. In the univariate analysis, a single SVR model is trained using the training data from WT53. The model predictions are then compared to the actual values, and the residuals (i.e., the differences between predicted and observed standard deviations) are used as an anomaly detection indicator.

In the multivariate analysis, four SVR models are trained, one for each of the four accelerometers installed at different positions in the nacelle (Fig. 4.4). these models are then used as a base for two anomaly detection techniques: the Mahalanobis distance and the One-Class Support Vector Machine (SVM).

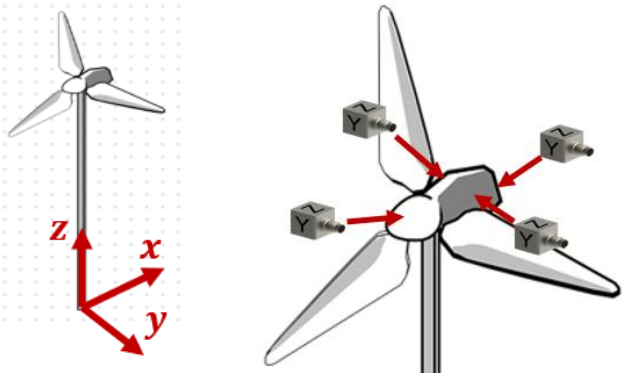

Figure 4.4: Approximate location of accelerometers and reference frame definition

Finally, the performance of the proposed anomaly detection methods is evaluated using various metrics commonly adopted in condition monitoring: accuracy, missed alarms, false alarms, F-score, and the ROC curve, from which the Area Under the Curve (AUC) is calculated.

In the remainder of this chapter, all the mathematical models used in this study will be described in detail. Some of them have been implemented directly through their defining equations, while others have been configured using built-in MATLAB functions by appropriately setting their parameters.

4.2 Betz Model

A simple yet fundamental model that describes how power output can be derived from wind energy input is the Betz model [41]. Albert Betz, an engineer who earned his PhD at the University of Göttingen, a key centre for the development of aerodynamics, proposed a physical framework that abstracts from the technological details of wind turbines. In this model, a wind turbine is idealised as an actuator disc that generates a pressure discontinuity when impacted by a flowing fluid, thereby extracting energy from the wind.

This analysis is based on the following idealised assumptions:

- The fluid flow is homogeneous, incompressible, and steady
- There is no frictional drag
- The turbine has an infinite number of blades
- Thrust is uniformly distributed over the disc or rotor area
- The wake is non-rotating
- Static pressure far upstream and downstream of the rotor is equal to the undisturbed ambient pressure

The core concept is that the turbine extracts kinetic energy from the wind by reducing its velocity, which in turn causes the flow tube to expand, as illustrated in Fig. 4.5.

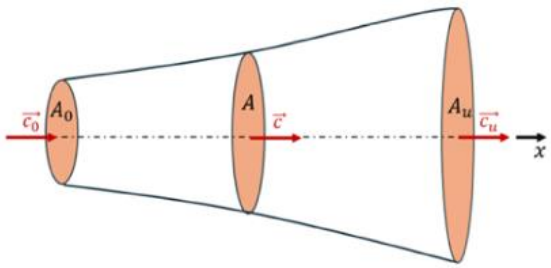

Figure 4.5: Representation of the actuator disc generating a pressure discontinuity for power extraction

Starting from the definition of the theoretical power output P_t of a wind turbine, given by (4.1):

$$P_t = \frac{1}{2}\dot{M}(c_0^2 - c_u^2) \tag{4.1}$$

Where P_t is the power output, $\dot{M}$ is the mass flow rate, c_0 is the upstream wind speed and c_u is the downstream wind speed. In line with the scientific practice of that era, when numerical calculators were not yet available, equation (4.1) is reformulated using a dimensionless variable known as the axial induction factor (or interference factor), defined as:

$$a = \frac{1}{2}\left(\frac{c_0 - c_u}{c_0}\right) \tag{4.2}$$

Combining equations (4.1) and (4.2), one obtains:

$$P_t = 2\rho A c_0^3 a(1-a)^2 \tag{4.3}$$

Thus, power output is expressed as a function of both the axial induction factor and the cube of the wind speed c_0, as well as the rotor area A.
In addition, another dimensionless parameter is introduced: the power coefficient C_p, which quantifies the rotor performance in terms of power extraction efficiency, defined as:

$$C_p = \frac{P_t}{\frac{1}{2}\rho A c_0^3} = 4a(1-a)^2 \tag{4.4}$$

From this expression, the power output can also be rewritten as:

$$P_t = \frac{1}{2}\rho A C_p c_0^3 \tag{4.5}$$

In this model, the power coefficient C_p represents the fraction of kinetic wind energy that can be extracted by the rotor. When expressed purely in terms of the axial induction factor, C_p forms a quadratic function that reaches its maximum at the same point as the power output. This ideal behaviour of a wind turbine is depicted in Fig. 4.6.

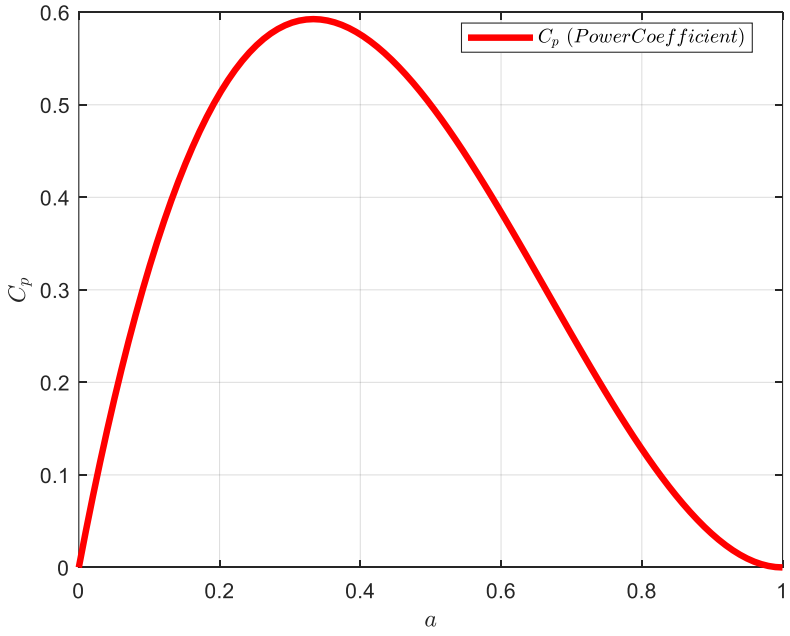

Figure 4.6: Power Coefficient C_p as a function of the axial induction factor

The model is not valid for values of $a > 0.5$, since this would imply $c_u > c_0$, which is physically unrealistic. The Betz limit, therefore, corresponds to the maximum achievable value of the power coefficient:

$$C_p = 0.59 \tag{4.6}$$

This represents the theoretical upper limit of power extraction efficiency for an ideal rotor.

In this thesis, as illustrated in Fig. 4.1, two distinct approaches were adopted:

- In a first analysis, equation (4.5) was used by fixing all the physical variables expect for wind speed, which comes from the measured data and elevated to the third power
- In a second analysis, a third-order polynomial regression model was developed to estimate the coefficients of equation (4.5), based on real wind turbine power output data, in order to improve regression analysis and anomaly detection performances

The main objective of these analyses was to improve the predictive capabilities of the SVR model and thereby enhance the performance of the anomaly detection framework.

In the following section, a general formulation of linear regression will be presented. This model was first used in the statistical analysis and subsequently adapted to a non-linear polynomial regressor for estimating the coefficients in equation (4.5).

4.3 Linear Regression Models

This section presents two distinct regression approaches adopted within the proposed methodology, each serving a specific purpose in modelling the wind turbine system.

First, a general statistical regression is employed to explore the relationship between the turbine's operating variables and output quantities such as nacelle vibration and power production. This analysis provides insights into the most relevant parameters under both normal and faulty operating conditions. Additionally, a Quantile-Quantile plot (QQplot) is applied to the residuals to verify the normality assumption required for subsequent application of the Mahalanobis distance, which relies on normally distributed data.

Second, a regression model based on the Betz equation is developed specifically to model the relationship between wind speed and power output. Two formulations of the Betz-based model are considered: the traditional theoretical approach, which assumes power output is proportional to the cube of wind speed according to (4.5), and a more flexible data-driven approach, where coefficients $\{\beta_i\}$ associated with different wind speed features are estimated via non-linear regression directly from the dataset. These estimated

coefficients are then used to enrich the input matrix of a Support Vector Regression (SVR) model by multiplying them with their corresponding wind speed features, thus embedding a data-refined physical model within the SVR training process.
The two Betz-based formulations are compared in terms of their impact on the SVR model's performance, specifically evaluating whether data-driven coefficients estimation improves prediction accuracy and anomaly detection compared to relying on fixed theoretical assumptions. This comparison allows a critical assessment of whether simple idealised models or tailored data-informed relationships better capture the turbine's actual behaviour.

As introduced in previous chapters, linear regression is a fundamental tool for the initial statistical analysis of a MISO system, enabling the study of the relationships between multiple physical parameters and a single dependent variable, such as the wind turbine's power output. Alongside this, a third-order polynomial regression model is employed to estimate the electrical power generated by the turbine, based on the Betz equation.
Let $\{y\} \in \mathbb{R}^{Nx1}$ denote the output vector (target or dependent variable), and $[X] \in \mathbb{R}^{NxM}$ the input matrix of predictors (independent variables), where N is the number of observations, and M is the number of features. Using matrix notation, the variables involved in a linear regression model are defined as follows:

$$[X] = \begin{pmatrix} 1 & x_{12} & \dots & x_{N1} \\ \vdots & \vdots & x_{ij} & \vdots \\ 1 & x_{N2} & \dots & x_{NM} \end{pmatrix}, \qquad \{y\} = \begin{pmatrix} y_1 \\ \vdots \\ y_N \end{pmatrix}, \qquad \{\beta\} = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_M \end{pmatrix}, \qquad \{\varepsilon\} = \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_N \end{pmatrix}$$

The linear regression model is defined as:

$$\{y\} = [X]\{\beta\} + \{\varepsilon\} \tag{4.7}$$

The regression function expresses the expected relationship between the target and the predictors, which is generally unknown. Thus, the main objective is to estimate this relationship by determining the unknown parameter vector $\{\beta\}$. A standard approach to solving this model is the least squares method, in which the regression coefficients are estimated by minimising the error vector $\{\varepsilon\}$, i.e. the sum of squared residuals. The least squares cost function is given by:

$$S(\beta_0, \beta_1, \dots, \beta_M) = \{\varepsilon\}^T\{\varepsilon\} = (\{y\} - [X]\{\beta\})^T(\{y\} - [X]\{\beta\}) \tag{4.8}$$

To find the minimum, the partial derivatives with respect to the regression coefficients are set to zero:

$$\left.\frac{\partial S}{\partial \beta_i}\right|_{\beta_0,\beta_1,\ldots,\beta_M} = 0 \tag{4.9}$$

This leads to the normal equation:

$$[X]^T[X]\{\beta\} = [X]^T\{y\} \tag{4.10}$$

Solving for $\{\beta\}$ yields the least squares estimate:

$$\{\beta\} = ([X]^T[X])^{-1}[X]^T\{y\} \tag{4.11}$$

The predicted values are then obtained by:

$$\{\hat{y}\} = [X]\{\beta\} = [X]([X]^T[X])^{-1}[X]^T\{y\} = [H]\{y\} \tag{4.12}$$

Here, the matrix $[H]$ is defined as:

$$[H] = [X]([X]^T[X])^{-1}[X]^T$$

The residuals, representing the differences between the predicted and actual values, are computed as:

$$\{r\} = \{\hat{y}\} - \{y\} = ([I] - [H])\{y\} \tag{4.13}$$

These residuals provide a measure of how well the model fits the data. However, they do not directly quantify the prediction error, which is why performance metrics such as the coefficient of determination R^2, mean absolute error MAE, and root mean square error $RMSE$ are also used. These are discussed in the following sections.

As illustrated in Fig. 4.7 [44], a typical example of a two-input, single-output linear regression is shown. This example depicts a regression plane fitted to the data points, where the vertical lines represent the residuals as defined in equation (4.13). The sum of the squared lengths of these lines is minimised in the least squares approach.

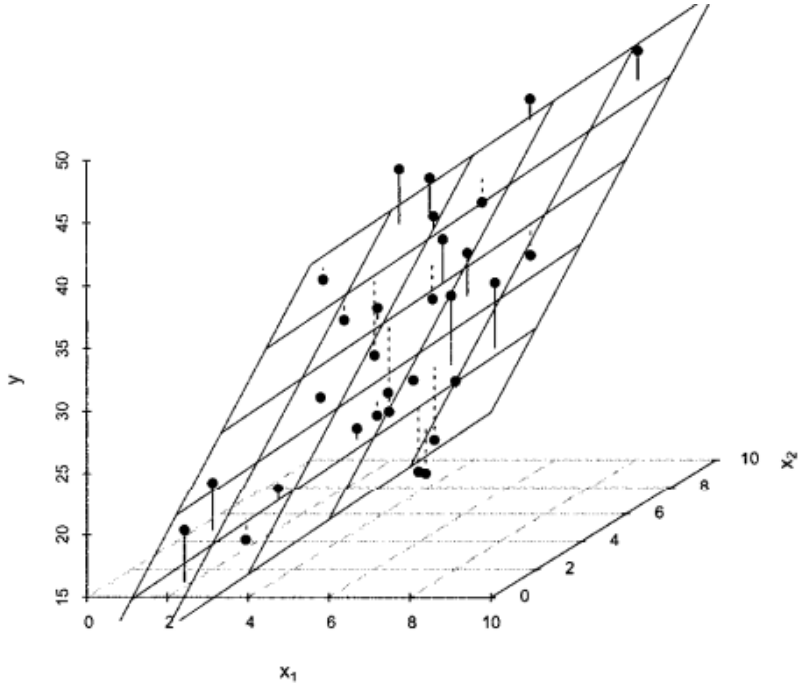

Figure 4.7: Example of a MISO regression model with two predictors

In the second part of this analysis, a third-order polynomial regression model is applied to evaluate the physical coefficients related to equation (4.5), derived from the Betz model. The objective is to estimate the output power generated by the turbine, considering wind speed as the main input. Based on the reference equation, the third-order regression model is formulated as:

$$\{P_t\} = \{\beta_0\} + \{\beta_1\}[c_0] + \{\beta_2\}[c_0^2] + \{\beta_3\}[c_0^3] \tag{4.14}$$

Here, $\{\beta_i\} \in \mathbb{R}^{Nx1}$ are regression coefficient vectors, and $[c_0] \in \mathbb{R}^{Nx4}$ represents the wind speed matrix, whose four columns correspond to extracted features: maximum, minimum, average, and standard deviation of the wind speed.

The coefficients $\{\beta_i\}$ derived from this model are used to improve the SVR by multiplying them with their corresponding wind speed features within the input matrix. Specifically, the resulting features are then added to the input matrix, which also includes other turbine operating parameters. This approach allows the integration of Betz-based physical insights by approximating the wind power transmitted as excitation to the dynamical system.

Both the linear statistical regression and the non linear Betz-based regression are independently evaluated using performance metrics such as MAE and $RMSE$ to assess their effectiveness in capturing the system's underlying behaviour.

4.4 Principal Component Analysis

After conducting a comprehensive statistical analysis and understanding the operational behaviour of the wind turbine and its correlation with various physical parameters, the input matrix of the wind turbine was constructed. Specifically, all the features derived from the variables discussed in Chapter 3 were included. The final input dataset, incorporating all relevant statistical descriptors, resulted in a 32-dimensional matrix:

$$[X] \in \mathbb{R}^{Nx32}$$

However, working with such a high-dimensional matrix can be computationally intensive, especially during the training of the SVR model. To address this, PCA was applied to reduce the dimensionality of the input space while retaining most of the original information through variance preservation.

PCA is a classic technique introduced by Karl Pearson in 1901 to determine the best-fitting lower-dimensional subspace that captures the structure of a dataset in a high-dimensional Euclidean space. Over the years, it has been widely used across various fields. Fundamentally, PCA transforms a dataset composed of potentially correlated variables into a new set of uncorrelated variables known as Principal Components (PCs) [24]. This transformation is achieved through an orthogonal rotation of the coordinate axes based on the directions of maximum variance.

Broadly speaking, PCA aims to identify a lower-dimensional subspace that effectively captures the essential structure of the original dataset while retaining as much information as possible. Practically, this means finding a linear transformation that projects the data into a new coordinate system, where the axes, defined by the PCs, are uncorrelated and ordered according to the variance they capture. The first principal component corresponds to the direction along which the data exhibit the greatest variance, and subsequent components are defined orthogonally to the previous ones, each capturing the next highest variance.

Mathematically, the linear transformation performed by PCA can be interpreted as a rotation of the original coordinate system. This rotation is determined by computing the eigen-decomposition of the covariance matrix of the input data. The resulting transformation matrix contains the eigenvectors of the covariance matrix as columns, sorted according to the magnitude of their corresponding eigenvalues in descending order. Thus, PCA is fundamentally an unsupervised learning technique that relies solely on the internal structure of the data.

Before applying PCA, it is essential to normalise the dataset. Several approaches can be found in the literature, such as subtracting the mean from each variable (centering approach). In this work, z-score normalisation has been adopted, defined as follows:

$$[X_{norm}] = \frac{[X] - \{\mu_{Train}\}}{\{\sigma_{Train}\}} \tag{4.15}$$

Here, $\{\mu_{Train}\}$ and $\{\sigma_{Train}\}$ denote the mean and standard deviation computed from the training set. These values are then used to normalise both the validation and test datasets, ensuring that no data leakage occurs across the phases.

At this point, the normalised input matrix $[X_{norm}]$ consists of N observations and 32 features and is ready for PCA and for training the SVR model. The covariance matrix of the normalised data is computed using the standard formula:

$$[\Sigma] = \frac{1}{N-1} \cdot [X_{norm}][X_{norm}]^T \tag{4.16}$$

PCA is then carried out by solving the following eigenvalue problem:

$$[\Sigma][V] = [V][\lambda] \tag{4.17}$$

In this expression, $[V]$ is an orthogonal matrix, whose columns are the eigenvectors v_j, while $[\lambda]$ is a diagonal matrix containing the corresponding eigenvalues λ_j, arranged in decreasing order of magnitude.

The matrix of eigenvectors $[V]$ can be used as a linear transformation to decorrelate the dataset, effectively rotating the coordinate system toward the principal directions of the data:

$$[Z] = [V]^T[X_{norm}] \tag{4.18}$$

If the eigenvectors in $[V]$ are normalised (i.e., $v_i \cdot v_j = 1$), this transform is a pure rotation. In that case, the variance of each new variable z_j is equal to the corresponding eigenvalue λ_j:

$$\sigma_j^2 = var(z_j) = \lambda_j \tag{4.19}$$

Thus, the diagonal matrix $[\lambda]$ also represents the covariance matrix of the transformed dataset $[Z]$, confirming that this is the traditional PCA approach.

From a geometric standpoint, PCA can be linked to the representation of an ellipsoid centred at the origin, associated with any positive-definite matrix such as the covariance matrix $[\Sigma]$. This ellipsoid is defined by:

$$[X]^T[\Sigma]^{-1}[X] = 1 \tag{4.20}$$

The principal axes of this ellipsoid are aligned with the eigenvectors of $[\Sigma]^{-1}$, and the corresponding eigenvalues are the reciprocals of the squares of the semi-axis's length.

Since the eigenvectors of $[\Sigma]$ and $[\Sigma]^{-1}$ are the same, and the eigenvalues are reciprocal, the equation of the ellipsoid can be expressed in terms of the PCA-transformed data:

$$[X]^T[\Sigma]^{-1}[X] = [Z]^T[V]^T[\Sigma]^{-1}[V][Z] = \\ = [Z]^T[\lambda]^{-1}[Z] = \sum_j \frac{z_j^2}{\lambda_j} = 1 \tag{4.21}$$

This represents the equation of an ellipsoid whose semi-axes have lengths equal to:

$$\sqrt{\lambda_j} = \sigma_j$$

In this thesis, after solving the eigenvalue problem, it was necessary to determine the number of principal components required to retain at least 90% of the total variance. This was done by computing the explained variance (EV), which quantifies the proportion of the total variance captured by each principal component:

$$\pi_j = \frac{\lambda_j}{\sum_{i=1}^{N} \lambda_i} = \frac{\lambda_j}{tr([\Sigma])} \tag{4.22}$$

Here, $tr([\Sigma])$ denotes the trace of the covariance matrix. The cumulative explained variance over a subset N of components can be expressed as a percentage of the total:

$$\sum_{j \in N} \pi_j \times 100\%$$

In this work, retaining 90% of the total variance resulted in a reduction of the original 32-dimensional input space to only three principal components.

4.5 Support Vector Regression

SVR extends the classic SVM [45], originally developed for classification problems, to the regression domain. As repeatedly emphasised throughout this thesis, the main objective is to train this model on data from a healthy wind turbine. This allows, through residual analysis (univariate), the development of a diagnostic framework for assessing the turbine's health status.

The maximal margin algorithm underlying SVR is based on a non-linear function that is learned using a linear learning machine in a kernel-induced feature space. The model complexity is controlled by a regularisation parameter that does not depend on the

dimensionality of the space. As in classification problems, the learning algorithm minimises a convex functional, and the resulting solution is sparse.
A key feature of SVR, similar to SVM, is the use of a loss function that ignores errors within a certain margin from the true target value. This is known as the ε-insensitive loss function, illustrated in Fig. 4.8.

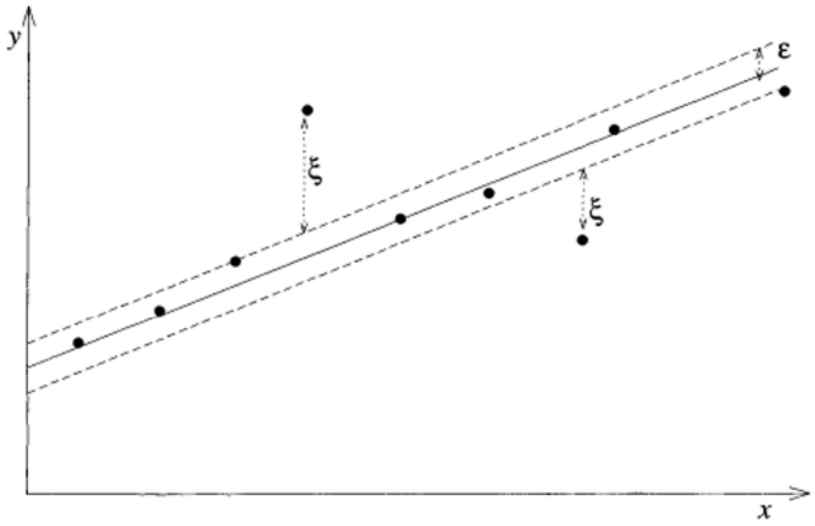

Figure 4.8: Example of an ε-insensitive band for a one-dimensional linear regression problem

In Fig. 4.8, the variables ξ are introduced to quantify the cost of errors on training points that fall outside the ε-band. These values are zero for all points within the band. For many reasonable choices of the loss function, the solution corresponds to the minimum of a convex functional.

This section presents the mathematical formulation of the regression method implemented in MATLAB through a dedicated function, along with a general overview of the parameters that must be tuned to achieve optimal regression performance. SVR, like SVM, is formulated through a Lagrangian function, whose minimisation leads to the so-called Support Vector Expansion in kernel space. Specifically, SVR is solved through its dual formulation, where the following variables are introduced:

$$\alpha_i^{(*)} = \{\alpha_i, \alpha_i^*\}$$

The mapping into the kernel space is expressed as:

$$\phi: X \to F$$

where F is the feature space. According to the Boser and Guyon approach, only the inner product is required:

$$k(x, x') = \langle \phi(x), \phi(x') \rangle$$

Furthermore, the Lagrangian for SVR is defined as:

$$\mathcal{L} = -\frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i - a_i^*)\left(\alpha_j - \alpha_j^*\right)k\left(x_i, x_j\right) - \varepsilon\sum_{i=1}^{N}(\alpha_i + \alpha_i^*) + \\ + \sum_{i=1}^{N} y_i(\alpha_i - \alpha_i^*) \tag{4.23}$$

subject to the constraints:

$$\sum_{i=1}^{N}(\alpha_i^* - \alpha_i) = 0\,, \qquad \alpha_i^{(*)} \in [0, C] \tag{4.24}$$

So that, the support vector weight is easily computed:

$$w = \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)\phi(x_i) \tag{4.25}$$

And finally, the predicted output is given by:

$$f(x) = \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)k(x_i, x) + b \tag{4.26}$$

As in SVM, it is unnecessary to compute a large set of basis functions $f_1(x), f_2(x), \ldots, f_N(x)$. Instead, only the kernel function $k(x_i, x)$ needs to be evaluated at the training and prediction points.

In this work, a Gaussian kernel is used, defined as:

$$k(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right) = \exp(-\gamma\|x - x'\|^2) \tag{4.27}$$

The kernel function maps the data into a high-dimensional feature space, where the only required operation is the inner product.

After selecting an appropriate kernel, here the Gaussian kernel, which is widely used in this context-several parameters must be tuned to optimise the model performance. These include:

- γ: the kernel scale parameter
- C: the box constraint, which controls the trade-off between model complexity and training error
- ε: defines the margin of tolerance, meaning the SVR model attempts to fit the data within a deviation of ε from the true values without penalising those deviations.

From a computational perspective, SVR can be implemented using different solvers. In this work, Sequential Minimal Optimisation (SMO) is employed, a commonly used algorithm for solving SVM problems. SMO breaks the optimisation problem into a series of smaller subproblems involving only two variables at a time. At each iteration, Lagrange multipliers are selected and solved analytically using the method described in Appendix.

Moreover, given the high computational cost of SVR, a reduced input matrix is used during the training phase to reduce runtime. As with the regression models used during the preprocessing phase, performance is continuously monitored using standard evaluation metrics, which are detailed in the following section.

In Table 4.3 is summarised the parameter values used to control the training phase of the SVR model.

Table 4.3: Values of SVR model parameters used during analysis

$\boldsymbol{\varepsilon}$	$\boldsymbol{C}$	**Num. Iterations**	$\boldsymbol{\gamma = \sigma\sqrt{2}}$	**Solver**
0.01	1	25500	15	SMO

4.6 Regression Metrics Performance

After training the SVR model using data collected under healthy operation conditions of the wind turbine, the same model was validated by applying it to a set of data that had never encountered before. The evaluation of its predictive capability can be performed through various approaches; however, for this analysis, the following statistical metrics were selected due to their widespread use and interpretability:

- Coefficient of determination R^2:

  The coefficient of determination, typically indicated as R^2, is a statistical index used to assess how well the predicted values align with the actual data. It represents the proportion of the total variance in the observed values that can be explained by the model. This metric is calculated by first determining the Sum of Squared Errors (SSE), which quantifies the discrepancy between the predicted and the measured values, and the Total Sum of Squares (SS), which expresses the total variability present in the target variable.

$$SSE = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 , \qquad SS = \sum_{i=1}^{N} \left(y_i - \mu_y\right)^2$$

The coefficient is then computed as:

$$R^2 = 1 - \frac{SSE}{SS} \tag{4.28}$$

where $\hat{y}_i$ represents the predicted value obtained from the model, y_i the actual measured value, and μ_y the mean of the measured values. The values of R^2 lies in the interval $(-\infty, 1]$, where 1 represents a perfect prediction. Thus, higher values of R^2 indicate a better fit between the model predictions and actual data.

- Root Mean Square Error (RMSE):

  The RMSE is another widely adopted metric and is defined as the square root of the Mean Squared Error (MSE). It provides an estimation of the standard deviation of the prediction errors, assuming they are normally distributed. The RMSE is computed as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2} \tag{4.29}$$

  This metric is particularly sensitive to larger errors, making it a robust choice when such deviations need to be emphasised in the performance assessment.

- Mean Absolute Error (MAE):

  The MAE measures the average absolute differences between predicted values and actual observations. It is the arithmetic mean of the absolute residuals and is expressed as:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i| \tag{4.30}$$

  This index can be interpreted as the average magnitude of the errors without considering their direction. MAE is conceptually related to Manhattan distance and is generally appropriate when the error distribution can be assumed to be normal. To verify this assumption, a QQplot of the residuals is also generated during statistical analysis. The QQplot allows for visual comparison of the residual distribution against a theoretical normal distribution. If the plotted points closely follow the reference line, it confirms the validity of applying MAE in the evaluation.

All the metrics discussed above provide different perspectives on the regression performance. In particular, RMSE is not bound to any specific assumption about the error

distribution, making it a more universally applicable metric. Conversely, MAE becomes more informative when the residuals exhibit a Gaussian distribution. Using both RMSE and MAE together, along with R^2, enables a comprehensive and balanced assessment of the model's ability to generalise the underlying physical behaviour, especially with regard to the nacelle's vibration dynamics.

4.7 Anomaly detection methods

This section outlines the methodology adopted for selecting and implementing tools for anomaly detection. As depicted in Fig. 4.1, two distinct strategies were employed and subsequently compared.
The first approach is a univariate analysis in which a single SVR model is trained. The input matrix is first reduced using PCA, as described in Equation (4.18), and the model is used to estimate the standard deviation of vibration signals (Fig. 4.9).

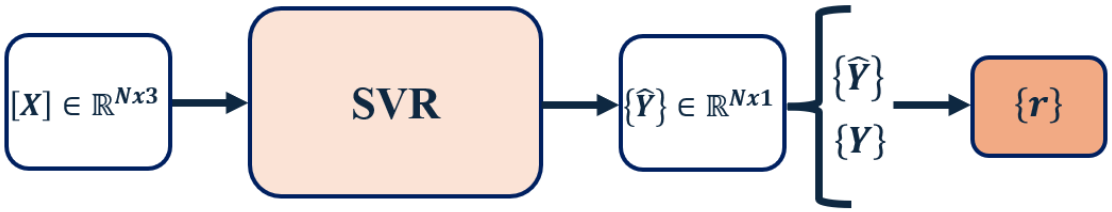

Figure 4.9: Illustration of the univariate anomaly detection process using a single SVR model

During the testing phase, the same trained model is applied to a dataset containing a pitch-failure event (refer to Table 4.1), and the residuals, chosen as anomaly indicator, are calculated according to Equation (4.31):

$$r_i = \hat{y}_i - y_i \tag{4.31}$$

This residual reflects the difference between the predicted and actual vibration standard deviation. A significant deviation may indicate the presence of a fault, as the SVR model fails to generalise due to abnormal operating conditions. However, such discrepancies might not exclusively signify damage, they could also result from external environment changes such as wind speed or temperature fluctuations. Furthermore, the univariate approach, relying on a single output signal, may be prone to a higher rate of false positives. Nevertheless, leveraging the standard deviation of the vibration signal, which captures its energy content, can enhance robustness to such confounding factors.

In addition to residuals analysis, the statistical distribution of the healthy and faulty dataset is compared. This comparison aids in determining a threshold value for the residuals. Any test data point with a residual exceeding this threshold can be classified as anomalous.

The second method, designed to offer a more comprehensive and reliable detection scheme, involves a multivariate analysis. In this case, four separate SVR models are trained simultaneously, as shown in Fig. 4.10.

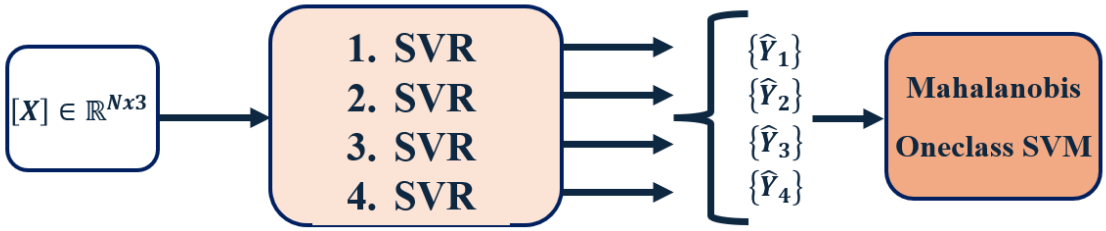

Figure 4.10: Multivariate anomaly detection approach using four trained SVR models

This method enables the use of more advanced anomaly scoring techniques. In particular, two techniques were selected in this work: the Mahalanobis Distance (MD), and the One-Class Support Vector Machine (OCSVM). Both methods are designed to produce a scalar score capable of distinguishing between healthy and faulty operating conditions.
The MD, introduced by Indian statistician P.C. Mahalanobis, is a statistical measure used for outlier and anomaly detection. Unlike Euclidean distance, which simply calculates the straight-line distance between two points, MD accounts for the correlations between variables. Given a multivariate data point $[X]$, a mean vector $\{\mu\} = E([X])$, and a covariance matrix $[S]$, as defined in (4.16), the Mahalanobis distance is defined as:

$$D([X], \{\mu\}) = \sqrt{([X] - \{\mu\})^T [S]^{-1} ([X] - \{\mu\})} \tag{4.32}$$

If $[S]$ is the identity matrix, this metric simplifies to the Euclidean distance. In this work, the mean and covariance matrix are computed from the training residuals matrix. The Mahalanobis distance is then calculated on the test set residuals using these statistics. If the computed distance exceeds a pre-established threshold, the point is classified as anomalous.
Notably, MD is robust to quasi-linear confounding factors, making it a valuable method for separating anomalies from normal behaviour.

A second scoring method, used as an alternative to the MD, involves to the application of OCSVM. This approach, originally proposed by Schölkopf et al. (1999), was specifically designed to adapt the Support Vector Machine methodology to the one-class classification problem.

The core idea is to map the input data into a high-dimensional feature space through a kernel function and then treat the origin as the only member of a 'negative' class. The goal is to identify a decision boundary that separates the majority of data points (belonging to the 'normal' class) from the origin with maximum margin.

Formally, given a dataset with an underlying pdf $p(x)$, the problem is framed as identifying a simple subset S in the feature space such that the probability of a test point falling outside S is bounded by a predefined value $v \in (0,1)$. In other words, points lying outside S are considered as anomalies.

The OCSVM algorithm aims to learn a decision function $f(x)$, which outputs +1 for inputs that lie within the learned boundary (i.e., inside S), and -1 for those outside it (i.e., in the complement $\bar{S}$ [46]:

$$f(x) = \begin{cases} +1 & if\ x \in S \\ -1 & if\ x \in \bar{S} \end{cases} \tag{4.33}$$

A visual representation of this concept is shown in Fig. 4.11, where the data points are separated from the origin in the kernel-induced space.

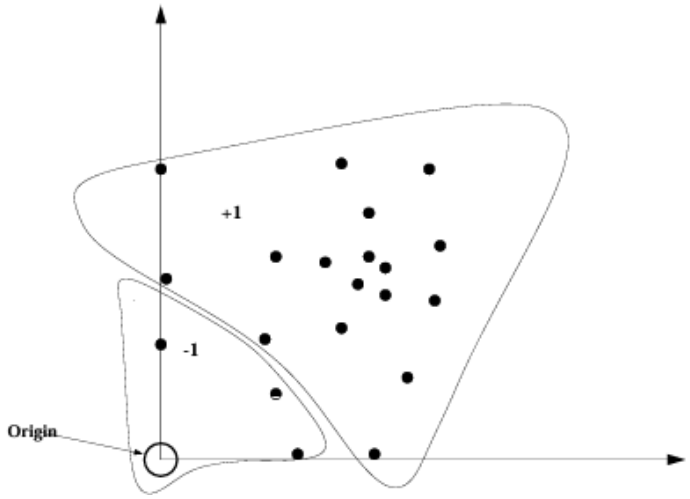

Figure 4.11: One-class SVM classification scheme. The origin is treated as the sole representative of the negative class

In this study, the residuals obtained from the training phase of the regression models are used as input for training the OCSVM. Let $\phi: X \to H$ denote the kernel map that transforms the residuals vectors into the feature space H. The inner products between transformed data points are computed using a Gram matrix, defined as:

$$G(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle \tag{4.34}$$

In the present work, a linear kernel is adopted, so the Gram matrix simplifies to:

$$G(x_i, x_j) = x_i^T x_j$$

The OCSVM problem is then formulated as the following quadratic optimisation problem:

$$\min \frac{1}{2}\|w\|^2 + \frac{1}{vN}\sum_{i=1}^{N}\xi_i - \rho \tag{4.35}$$

Subject to the constraints:

$$\left(w^T \cdot \phi(x_i)\right) \geq \rho - \xi_i, \qquad i = 1,2,\dots,N, \;\; \xi_i \geq 0 \tag{4.36}$$

Here, w and ρ define the decision boundary, while ξ_i are slack variables allowing for some margin of error in classification. Once the optimisation problem is solved, the resulting decision function is computed:

$$f(x) = sign\left(\left(w \cdot \phi(x_i)\right) - \rho\right) \tag{4.37}$$

The decision function returns +1 for most of the training samples, which lie within the region S, and -1 for those identified as outliers. This makes OCSVM a powerful tool for detecting anomalies in systems where only data from normal operation are available during training.

4.8 Classification models adopted for multi-class analysis

The final part of this work is based on the implementation of a multi-class analysis involving two different methods for classifying both healthy and faulty conditions. In this framework, four turbines from the same wind farm have been considered, each one affected by a different type of anomaly related to a fault condition. The goal is to assess the model's ability to distinguish between different types of damage during classification. This evaluation is carried out by means of a specific performance metric, namely the Class Error, computed from the confusion matrix, as discussed in the following paragraph.

The final analysis is performed using the residuals obtained from the trained SVR model. Specifically, the residuals of the vibration data resulting from the multivariate analysis, identified as the most accurate model as discussed in the next chapter, are considered. A label vector is built by distinguishing between the normal operating condition and the different types of damage.

Although the process of assembling the final dataset from the SVR output and the structure of the associated labels is discussed in the next chapter, this paragraph focuses on the mathematical formulation of the classification methods adopted in this work. In particular, two classification techniques have been implemented:

- LDA: Linear Discriminant Analysis
- QDA: Quadratic Discriminant Analysis

In addition, a cross-validation procedure has been applied. The constructed dataset is divided into k-folds: for each iteration, one-fold is used as the validation set, while the remaining k-1 folds constitute the training set. This technique ensures that the entire dataset is used both for training and validation, and it allows to evaluate the generalisation capability of the classifiers.

The classification process is based on decision theory, which aims to determine the posterior probability $\Pr(Q|X)$ of a class Q given the observation X [21]. Let $f_i(x)$ denote the class-conditional density of X in class $Q = i$, and let ν_i be the prior probability of class i, where $\sum_{i=1}^{I} \nu_i = 1$. Applying Bayes' theorem, it is possible to obtain:

$$\Pr(Q = i|X = x) = \frac{f_i(x)\nu_i}{\sum_{l=1}^{I} f_l(x)\nu_l} \tag{4.38}$$

Both LDA and QDA are based on the assumption of Gaussian class-conditional densities, defined as:

$$f_i(x) = \frac{1}{(2\pi)^{p/2}|S_i|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i)\right) \tag{4.39}$$

LDA corresponds to the particular scenario in which all classes are assumed to share the same covariance matrix, denoted as $\Sigma_i = \Sigma$ for each class i. Under this condition, the classification rule between any two classes, i and l, can be derived by analysing the log-ration of their posterior probabilities. By applying Bayes' theorem under the assumption of Gaussian class-conditional densities, this log-ratio simplifies to:

$$\log\left(\frac{\Pr(Q = i\,|X = x)}{\Pr(Q = l|X = x)}\right) = \log\frac{\nu_i}{\nu_l} - \frac{1}{2}(\mu_i + \mu_l)^T\Sigma^{-1}(\mu_i - \mu_l) + \\ +x^T\Sigma^{-1}(\mu_i - \mu_l) \tag{4.40}$$

This expression is linear in x, as the shared covariance matrix causes both the normalisation terms and the quadratic components in the exponentials to cancel. As a result, the decision boundary between any two classes is defined by a linear function, and,

in a p-dimensional feature space, this corresponds to a hyperplane. Consequently, all classes boundaries in LDA are linear and partition the space $\mathbb{R}^p$ into regions separated by hyperplanes (Fig. 4.12).

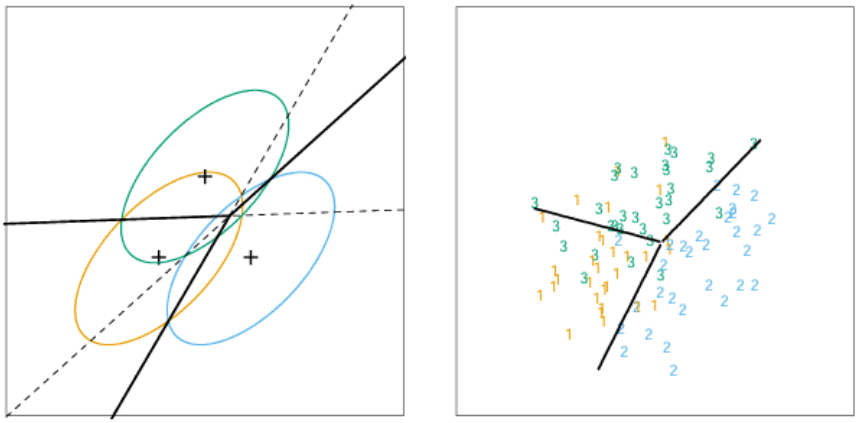

Figure 4.12: Left-Three Gaussian distribution are shown, each characterised by a different mean but sharing a common covariance structure. The dashed lines denote the optimal pairwise decision surfaces derived from Bayes theory, while the solid lines represent the overall multi-class separation regions. Right-A sample of 30 observations per distribution is presented, alongside the linear boundaries estimated through LDA

From equation (4.40), it is possible to define the linear discriminant function for each class i as:

$$\delta_i(x) = x^T\Sigma^{-1}\mu_i - \frac{1}{2}\mu_i^T\Sigma^{-1}\mu_i + \log \nu_i \tag{4.41}$$

Since the true parameters ν_i, μ_i and Σ are unknown in practice, they must be estimated from the training data. The standard estimators are:

- $\hat{\nu}_i = N_i/N$, where N_i is the number of samples in class i, and N is the total number of observations
- $\hat{\mu}_i = \sum_{j_n=i} \frac{x_i}{N_k}$ is the predicted mean for class i
- $\hat{\Sigma} = \sum_{i=1}^{K} \sum_{j_n=k} (x_i - \hat{\mu}_i)(x_i - \hat{\mu}_i)^T/(N-K)$ is the predicted within-class covariance matrix

In particular, a sample is assigned for instance to class 2 rather than class 1, when the following inequality is satisfied:

$$x^T\hat{\Sigma}^{-1}(\hat{\mu}_2 - \hat{\mu}_1) > \frac{1}{2}\hat{\mu}_2\hat{\Sigma}^{-1}\hat{\mu}_2 - \frac{1}{2}\hat{\mu}_1\hat{\Sigma}^{-1}\hat{\mu}_1 + \log\left(\frac{N_1}{N}\right) - \log\left(\frac{N_2}{N}\right) \tag{4.42}$$

The second classification method considered in this work is QDA. Unlike LDA, QDA does not assume that the covariance matrices are equal across classes. Each class i is associated with its own covariance matrix Σ_i. As a result, the quadratic term in the exponent of the Gaussian distribution is retained (4.39), leading to the following non-linear discriminant function:

$$\delta_i(x) = -\frac{1}{2}\log|\Sigma_i| - \frac{1}{2}(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i) + \log \nu_i \tag{4.43}$$

The decision boundary between any two classes i and l is thus described by a quadratic equation, defined by the set $\{x: \delta_i(x) = \delta_l(x)\}$. This allows QDA to capture more complex and curved decision boundaries, which may be more suitable in cases where class covariance structures differ significantly.

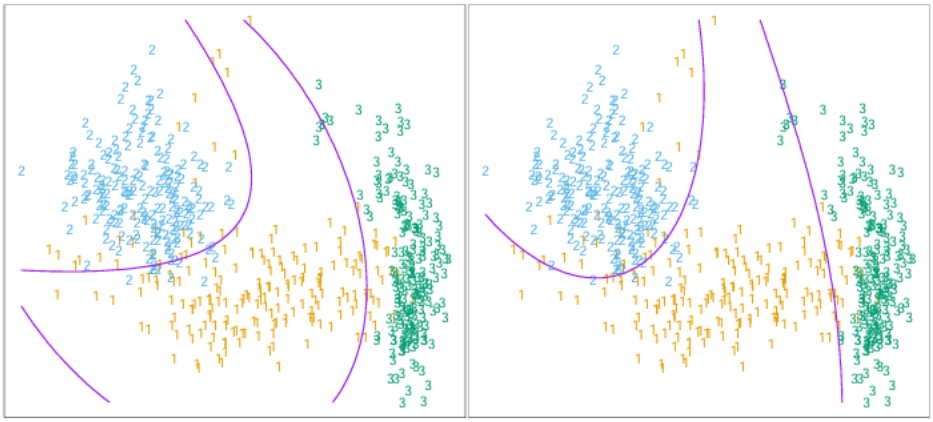

Figure 4.13: Comparison of classification boundaries generated by two different approaches. The plot on the left shows the result of applying LDA after expanding the input features through a quadratic transformation, leading to non-linear decision boundaries. The plot on the right illustrates the decision surfaces obtained with QDA, which naturally accommodates class-specific covariance structures. In both cases, the separation between the three classes is evident, although QDA provides inherently curved boundaries without the need for feature expansion

4.9 Metrics Performance Anomaly Detection

To conclude the analysis, it is important to assess whether the anomaly scores derived from the trained model are truly effective in identifying anomalous behaviour. This is typically done by evaluating how well the model performs when a threshold, determined based on the distribution of anomaly scores in the training data, is applied to distinguish between normal and anomalous instances.

A standard approach to such evaluation involves the use of the confusion matrix, which provides a structured summary of prediction outcomes in a binary classification context [48]. Specifically, it categorises the results as follows:

- True positives (TP): anomalous instances correctly identified as anomalies
- False positives (FP): normal points incorrectly flagged as anomalies
- False negatives (FN): anomalies that the model fails to detect
- True negatives (TN): normal points accurately classified as normal

These four values, also referred to as count-based metrics, form the basis for a range of evaluation indicators (Fig. 4.14), which are used in this thesis to quantitatively measure model performance.

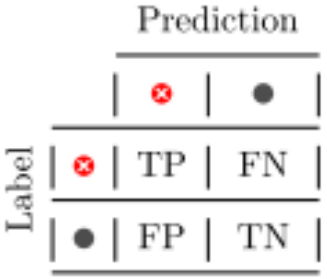

Figure 4.14: Confusion matrix representation

The first metric considered is Accuracy (ACC), which represents the proportion of all correct predictions (both anomalies and normal points) over the total number of samples:

$$ACC = \frac{TP + TN}{TP + TN + FN + FP} \tag{4.37}$$

While accuracy provides a general overview, it can be misleading in highly imbalanced dataset, which is often the case in anomaly detection.

To better capture the model's performance in identifying rare events, Recall (REC), also known as True Positive Rate or Sensitivity, is employed. Recall quantifies the proportion of actual anomalies that are successfully detected:

$$REC = \frac{TP}{TP + FN} \tag{4.38}$$

A high recall means that few anomalies go undetected, although it may come at the cost of increased false positives.
In contrast, Precision (P) focuses on the quality of the predictions made as anomalies, and is defined as the fraction of predicted anomalies that are actually anomalous:

$$P = \frac{TP}{TP + FP} \tag{4.39}$$

A high precision indicates that most anomaly alerts are valid, even if some actual anomalies are missed.

To strike a balance between these two aspects, the F-score is used. It is the harmonic mean of precision and recall, combining them into a single value:

$$F = \frac{2P \cdot REC}{P + REC} \tag{4.40}$$

Depending on the application, one might prioritise precision over recall (or vice versa), which can be adjusted through the F_β-score:

$$F_\beta = \frac{(1 + \beta^2)P \cdot REC}{REC + \beta^2 P} \tag{4.41}$$

The parameter β allows the analyst to express how much more importance should be given to recall relative to precision. For example, $\beta > 1$ favours recall, while $\beta < 1$ favours precision.

The metrics discussed so far are evaluated at a fixed threshold, but in many cases, one seeks to analyse how model performance changes across a range of thresholds. To this end, Receiver Operating Characteristics (ROC) analysis is particularly useful. The ROC curve is a plot of recall against the False Positive Rate (FPR) for varying threshold values, offering a global view of classification behaviour (Fig. 4.15).

To summarise the ROC performance in a single scalar, the Area Under the Curve (AUC) is computed. In this work, the AUC-ROC has been adopted as an additional indicator of the model's ability to discriminate between normal and anomalous conditions across all possible thresholds. A higher AUC value indicates more reliable and robust anomaly detection, and this metric has become a common standard in research for evaluating detection systems.

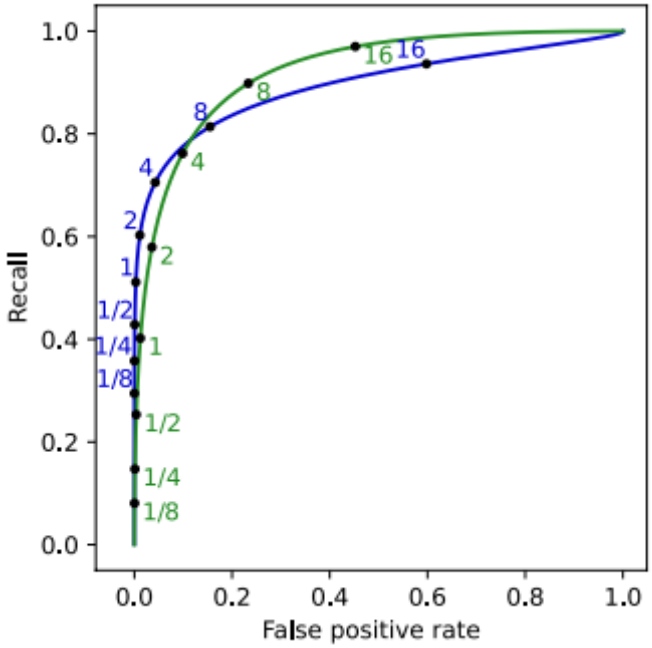

Figure 4.15: Example of a ROC curve comparing the performance of two anomaly detectors, labelled A (blue curve) and B (green curve). The plot illustrates the trade-off between true positive rate and false positive rate across different threshold values, highlighting the relative detection capabilities of each model

Additionally, to the binary class analysis conducted in this work, the final stage of this thesis focuses on performing a multi-class classification. In this framework, beyond the categories previously defined in the confusion matrix, an additional category, known as Class Error, is introduced (Table 4.4).

Table 4.4: Generic confusion matrix for multi-class classification, where the class k=0 corresponds to the healthy or reference state, while classes k=1… N represent different categories of damage

	True Class 0	True Class 1	True Class …	True Class k	True Class …	True Class N
Predicted Class 0	TN_0	FN_1	$FN_{...}$	FN_k	$FN_{...}$	FN_N
Predicted Class 1	FP_1	TP_1	$CE_{1,...}$	$CE_{1,k}$	$CE_{1,...}$	$CE_{1,N}$
Predicted Class …	$FP_{...}$	$CE_{...,1}$	$TP_{...}$	$CE_{...,k}$	$CE_{...}$	$CE_{...,N}$
Predicted Class k	FP_k	$CE_{k,1}$	$CE_{k,...}$	TP_k	$CE_{k,...}$	$CE_{k,N}$
Predicted Class …	$FP_{...}$	$CE_{...,1}$	$CE_{...}$	$CE_{...,k}$	$TP_{...}$	$CE_{...,N}$

Predicted Class N	FP_N	$CE_{N,1}$	$CE_{N,\dots}$	$CE_{N,k}$	$CE_{N,\dots}$	TP_N

Within this framework, the Class Error quantifies the proportion of samples that, although correctly detected as anomalous, have been misclassified into an incorrect damage category. This metric effectively measures the precision of the model in distinguishing among different fault types.
Formally, the Class Error Rate (CER) is defined as:

$$CER = \frac{CE}{TC} \tag{4.42}$$

Where CE corresponds to the number of misclassified samples located outside the main diagonal of the confusion matrix, and TC denotes the total number of samples classified. The CER thus provides a quantitative assessment of the model's ability to discriminate between multiple fault categories. This capability is particularly crucial in the context of continuous monitoring for wind farms, where numerous turbines must be supervised concurrently. In such scenarios, it is essential not only to differentiate normal operating states from anomalous conditions but also to accurately identify the specific type of damage affecting each turbine, enabling more targeted maintenance interventions and enhanced operational reliability.

A final performance metric is also computed in this work, to compare the classification results of QDA and LDA. This metric, referred to as the Performance Index (PI), combines the information provided by the accuracy, MA, FA and CER, allowing the evaluation to be condensed into a single scalar value, as defined in equation (4.43).

$$PI = ACC \cdot (1 - MA) \cdot (1 - FA) \cdot (1 - CER) \tag{4.43}$$

5. Results

As outlined in Chapter 4 and illustrated in Fig. 4.1, the methodology developed in this thesis follows a structured sequence of operations aimed at implementing a robust anomaly detection strategy for wind turbine monitoring. This chapter presents the results obtained from the application of the proposed models and provides a critical analysis of their effectiveness.

The first part of the chapter focuses on the preprocessing phase, which includes a statistical analysis of the dataset and a discussion of the potential influence of the Betz model introduced earlier. This section is intended to verify the consistency of the input data and to provide physical insight into the variables employed in the subsequent learning phase.

The second part is devoted to the regression models, with an emphasis on performance evaluation across the training, validation, and test sets. The metrics introduced in Chapter 4 are applied here to quantify the predictive capability of the models and to evaluate their robustness under varying operating conditions.

Following this, a first anomaly detection analysis is conducted using a univariate approach based on the residuals obtained from the regression model. Special attention is given to the interpretation of performance metrics, in order to assess the model's sensitivity in detecting deviations from normal behaviour.

The third section presents a multivariate analysis and compares it with the previous univariate model, highlighting differences in terms of predictive accuracy and diagnostic capability, as well as discussing the associated limitations and trade-offs.

Finally, a classification analysis is introduced. A cross-validation procedure is applied involving additional turbines, with the goal of assessing the model's ability to not only detect the presence of anomalies, but also to identify the specific type of fault, thus evaluating its generalisation potential.

5.1 Statistical Analysis

The preprocessing phase represents one of the most critical and delicate steps in the development of any machine learning framework, as it provides the necessary understanding of the dataset and establishes the foundation on which the entire methodology is built. As introduced in Chapter 3, a preliminary data cleaning procedure

was carried out to remove any irregularities or anomalies that could compromise the training process and, consequently, the accuracy of the model.

The first goal of this phase is to perform a statistical analysis aimed at identifying the main correlations between physical parameters and two key output variables: the nacelle vibration signal and the electrical power produced by the turbine. The power output is particularly relevant not only to gain a basic understanding of turbine behaviour, especially during the power control phase, but also to model the aerodynamic input applied by wind to the system, in terms of converted mechanical energy.

In all analyses where wind speed is considered, a cubic dependency has been explicitly included in accordance with the Betz theory. This formulation was used to approximate the aerodynamic power transmitted from the wind to the turbine, which acts as a physical input to the dynamic system. By incorporating this relationship, it is possible to estimate the amount of energy transferred from the airflow to the rotor and drivetrain, thus allowing for a more physically consistent modelling of the turbine's operating conditions.

The analysis initially focuses on identifying the best-fitting regression plane using a MISO regression. Various physical variables were selected from the dataset, excluding electrical signals in favour of mechanical ones, in line with the nature of the measured vibration signal. Several combinations of input variables were tested, and the RMSE and MAE were computed to evaluate the average deviation of the experimental data from the regression plane. In addition, QQ-plots were used to investigate the statistical distribution of residuals.

The most relevant results obtained through the regression analysis are shown in Fig. 5.1 to 5.5, where the vibration data is plotted against different combinations of physical parameters in three-dimensional spaces.

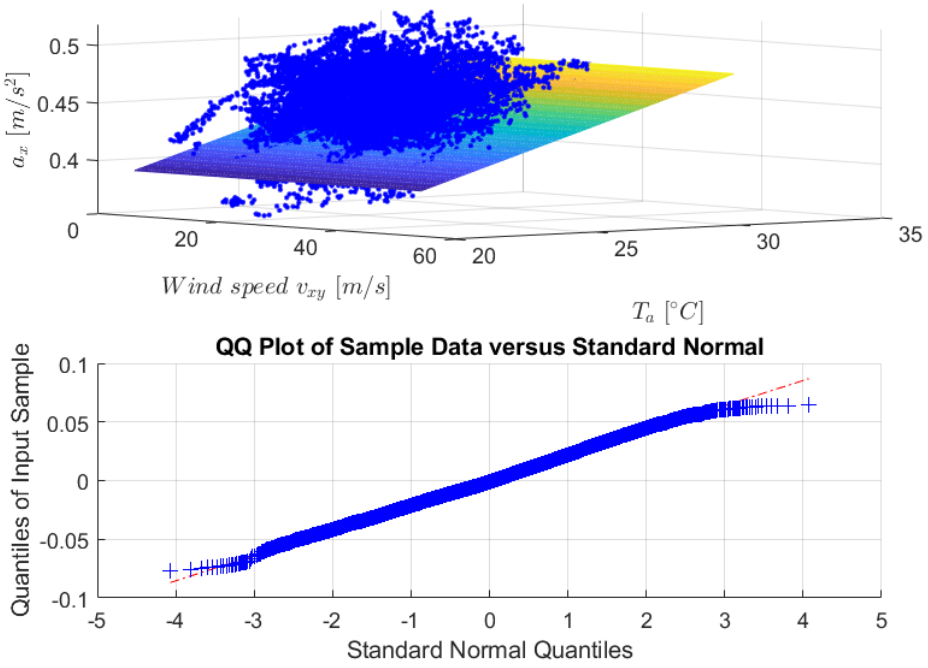

Figure 5.1: Acceleration vs wind speed and ambient temperature (top), and QQ-plot of residuals (bottom)

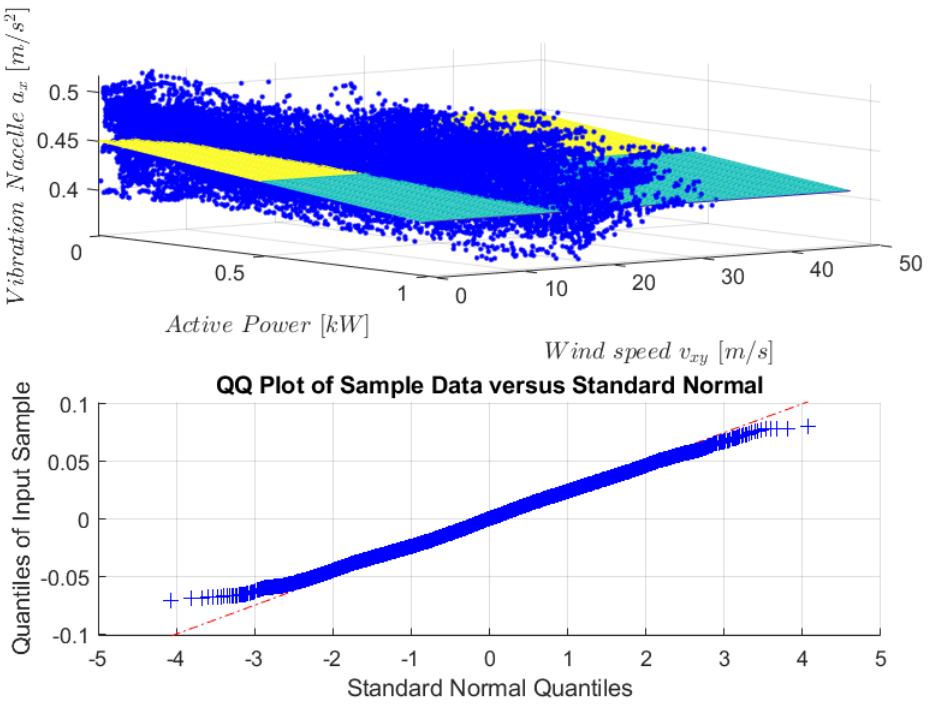

Figure 5.2: Acceleration vs power output and wind speed, QQ-plot residuals

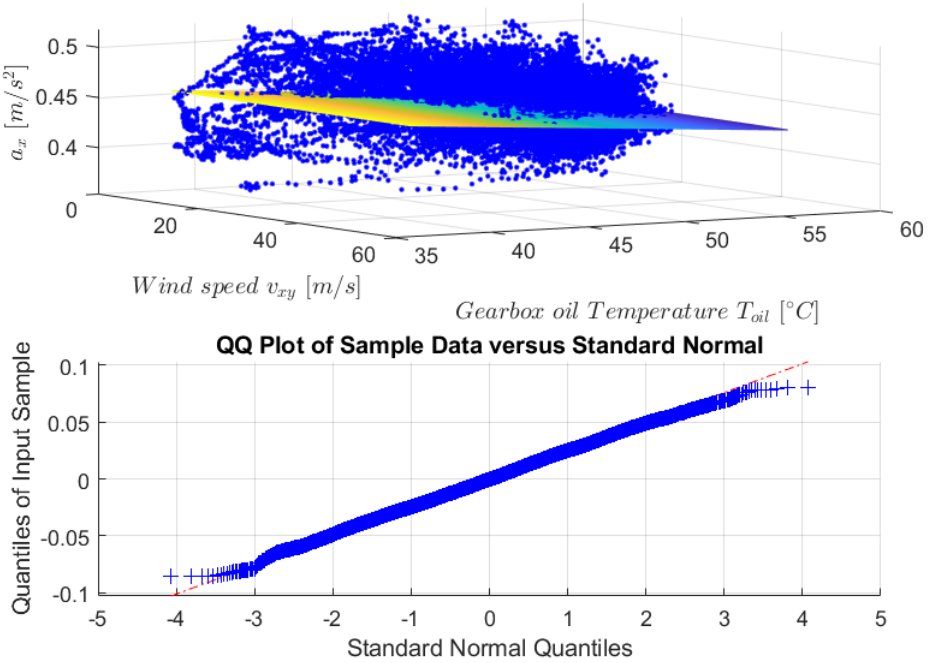

Figure 5.3: Acceleration vs wind speed and gearbox oil temperature, and QQ-plot residuals

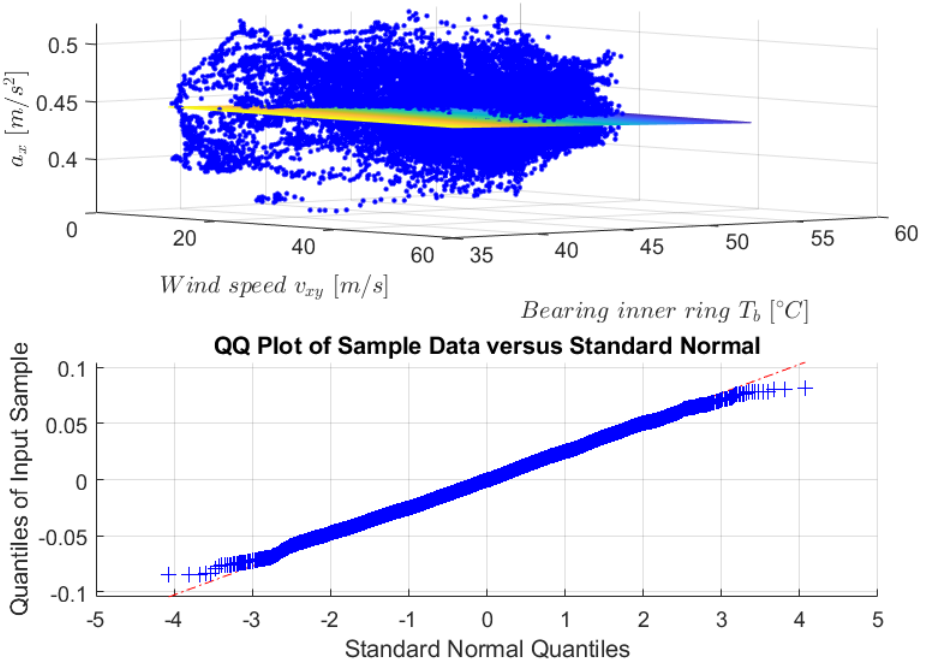

Figure 5.4: Acceleration vs wind speed and inner ring bearing temperature, and QQ-plot residuals

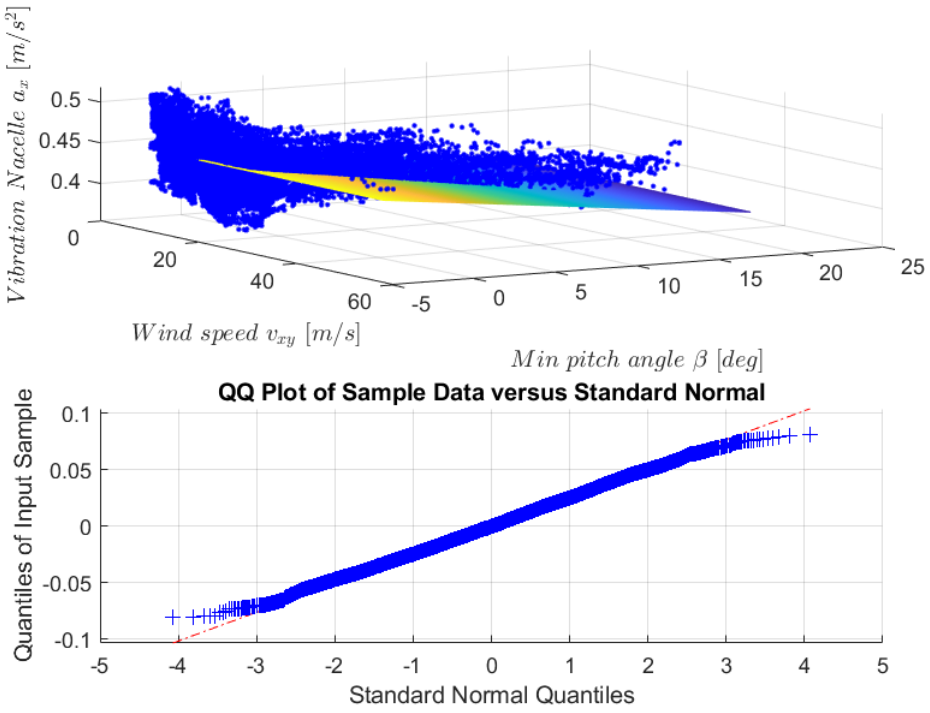

Figure 5.5: Acceleration vs wind speed and pitch angle, and QQ-plot residuals

Table 5.1: Summary of regression performance metrics (target: mean nacelle vibration signal)

	Plot 1	**Plot 2**	**Plot 3**	**Plot 4**	**Plot 5**
RMSE [%]	2.2	9.3	2.5	2.5	2.3
MAE [%]	1.7	7.5	2.00	2.00	1.9

Table 5.1 shows a good agreement between the regression plane and the experimental data, indicating a strong correlation between the nacelle vibration data and the selected physical parameters: power output, pitch angle, ambient temperature, wind speed, gearbox oil temperature, and inner ring bearing temperature. This conclusion is further supported by the error metrics (RMSE and MAE), which remain below 10% for all considered combinations. The values confirm the relevant and physical consistency of the selected input variables.

In particular, the use of the cubic dependency of wind speed, derived from the Betz model to approximate the aerodynamic power transmitted to the system, has proven to be effective. The results demonstrate that this assumption leads to accurate regressions, reinforcing the physical validity of this approach for modelling the wind's contribution to the turbine dynamics.

Moreover, the QQ-plot analysis highlights an additional aspect related to the statistical distribution of the residuals. Given the importance of residuals analysis in this work, particularly in the anomaly detection phase, it is observed that the residuals tend to follow

an approximately normal distribution, despite the slightly divergent tails. This behaviour may suggest the presence of nonlinear effects that are not fully captured by the linear regression model. The effectiveness of the data cleaning process is also evident: the removal of these non linearities has prevented distortion in both error metrics and residuals distributions, as shown in Fig. 5.6.

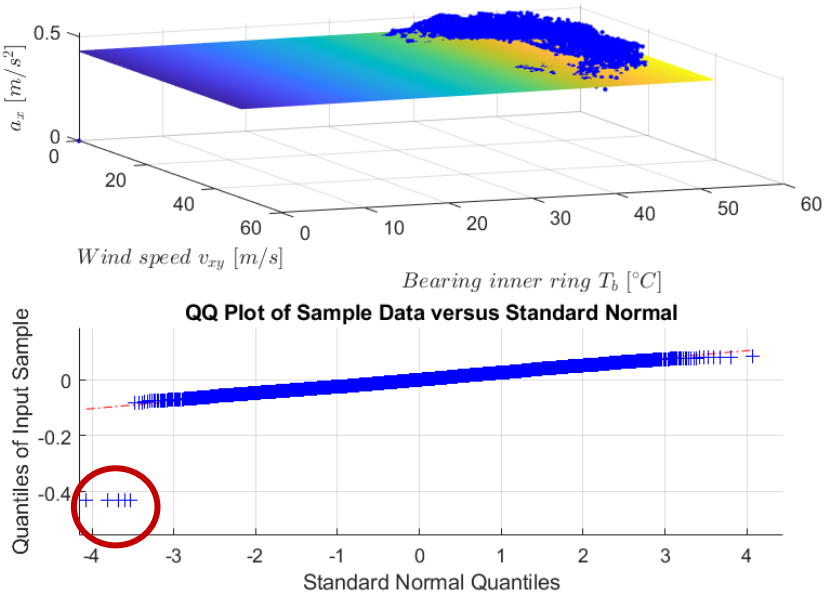

Figure 5.6: Acceleration vs wind speed and inner ring bearing temperature. Red circle indicate discontinuity caused by the presence of outliers in the dataset

From a physical standpoint, the following observations can be made regarding the turbine's behaviour under normal operating conditions:

- The turbine produces electrical power within a wind speed range of approximately $2 \div 28\ m/s$, corresponding to the cut-in and cut-off thresholds. Below or above this interval, the turbine does not generate power, as shown in Fig. 5.7. The maximum power output, reached at around $25\ m/s$, is approximately $1\ kW$, which classifies the turbine as a small-scale unit. This behaviour is confirmed by Fig. 5.2, where vibration data are absent beyond $30\ m/s$ of wind speed, indicating no data sampling was conducted above this limit.
- Fig. 5.2 further shows a decrease in nacelle vibrations as power output increases, a trend that coincides with the increase in wind speed, consistent with the Betz model. This highlights a key aspect of the turbine's power control strategy, which involves adjusting blade pitch to optimise aerodynamic efficiency and reduce turbulence-induced vibrations, thus limiting losses in aerodynamic performance.

Additionally, Fig. 5.6 reveals a reduction in vibration as the pitch angle increases. As discussed in Chapter 2, pitch failure is a common fault condition; based on these results, it can be expected that vibration amplitudes would increase when the turbine loses its capability to optimise its aerodynamic profile to meet the required power output. However, it should also be noted that stall control can significantly influence the vibration signals at the nacelle, and therefore a thorough cross-analysis is necessary before declaring a fault condition.

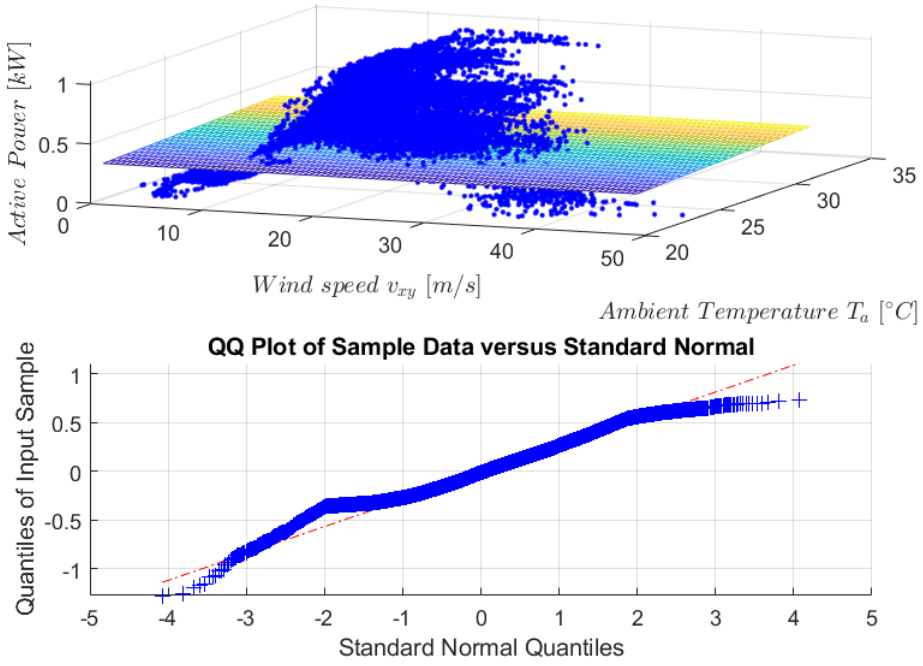

Figure 5.7: Power output vs wind speed and ambient temperature

- Fig. 5.3 and 5.4 show that, under normal operating conditions, the gearbox oil temperature and the inner ring bearing temperature of the main shaft remain within specific ranges. These parameters are crucial for condition monitoring, as even small temperature deviations may indicate malfunctions caused by phenomena such as wear or fatigue within the gearbox. Given their importance, these variables were included in the input matrix used during the SVR training phase. Consequently, during testing, anomalies detected in these parameters contribute to decreased predictive performance of the regressor, serving as an indicator of potential faults.

5.2 Univariate Analysis

A univariate analysis involves the development of a regression model in which a single output variable is predicted from a given set of input features. The main objective in this phase is to train the SVR model using an input matrix composed of the most influential physical parameters affecting the dynamic behaviour of the wind turbine, as reported in Table 5.2.

Table 5.2: Physical parameters used in SVR

Physical Variable	**Symbol**
Wind speed	c_0
Ambient Temperature	T_a
Pitch Angle	α
Gearbox Oil Temperature	T_{oil}
Rotor Blade Axis Position	β
Rotor Speed	ω_r
Bearing Inner Ring Temperature	T_{inb}
Rotor Speed Gearbox Main-shaft	ω_g

From the combination of these parameters and their statistical descriptors, i.e. maximum, minimum, mean and standard deviation, a matrix $[X] \in \mathbb{R}^{N\times 32}$ is constructed, where N is the number of observations. As previously emphasised throughout this thesis, particular attention is given to the physical interpretation provided by the Betz theory, as introduced in equation (4.5). In this context, wind speed is not only used as a direct feature but also transformed to reflect its cubic relationship with aerodynamic power, in line with fundamental energy conversion principles.

The rationale behind this transformation is to enhance the physical consistency of the input representation. Specifically, it allows the model to better approximate the aerodynamic power transferred from the wind to the turbine, an input that acts as a forcing term in the dynamic system and is closely linked to the vibration response observed at the nacelle. This assumption is strongly supported by the statistical analysis presented earlier,

which showed a clear correlation between nacelle vibration levels and wind speed, further highlighting the role of turbulence and unsteady aerodynamic effects in the excitation of structural vibrations.
The coefficient used in equation (4.5) to estimate wind power are summarised in Table 5.3.

Table 5.3: Coefficients assumed for aerodynamic power computation (Betz law)

Data	Values
$\rho\ [kg/m^3]$	1.23
C_p	0.40
$A\ [m^2]$	15

These parameters are selected based on the results of the previous analysis and relevant physical considerations. The air density ρ is assumed constant, as the wind is treated as an incompressible fluid. The power coefficient C_p is chosen according to the theoretical upper limit defined by the Betz law. The swept area A corresponds to that of a small-scale turbine, as justified by the observed maximum power output of approximately $1\ kW$. Furthermore, due to the low rotational speed of the rotor (and consequently of the blades), centrifugal effects and structural deformation are assumed to be negligible in this phase, reinforcing the assumption of moderate dynamic loads and stable operating conditions.
By combining these constants with the cube of the wind speed, a new feature is derived and included in the input matrix. This feature is intended to more accurately represent the dynamic excitation induced by wind energy, and to strengthen the link between the aerodynamic input and the turbine's dynamic response in terms of vibration.
Before proceeding with the training phase, a PCA was applied to reduce the dimensionality of the input matrix. This step is essential to lower the computational complexity, especially during the kernel transformation inherent to the SVR algorithm when projecting data into the higher-dimensional Hilbert space.

5.2.1 PCA Results

PCA was applied to reduce the dimensionality of the original input matrix, which includes 32 monitored features. This technique constructs new uncorrelated variables, known as principal components, as linear combinations of the original features, ordered by decreasing variance. The transformation maximises the retained variance and minimises

the redundancy and noise within the dataset by solving an eigenvalue decomposition of the covariance matrix.

Due to the high dimensionality of the original feature space, direct visualisation is not feasible. As introduced in Chapter 4, the cumulative explained variance curve was adopted to determine the optimal number of principal components to retain. Fig. 5.8 shows the proportion of total variance explained as a function of the number of components. A threshold of 90% was selected, which balances dimensionality reduction with information preservation. This led to the retention of the first three principal components, effectively projecting the original 32-dimensional input space onto a 3-dimensional subspace, while maintaining the majority of its informational content.

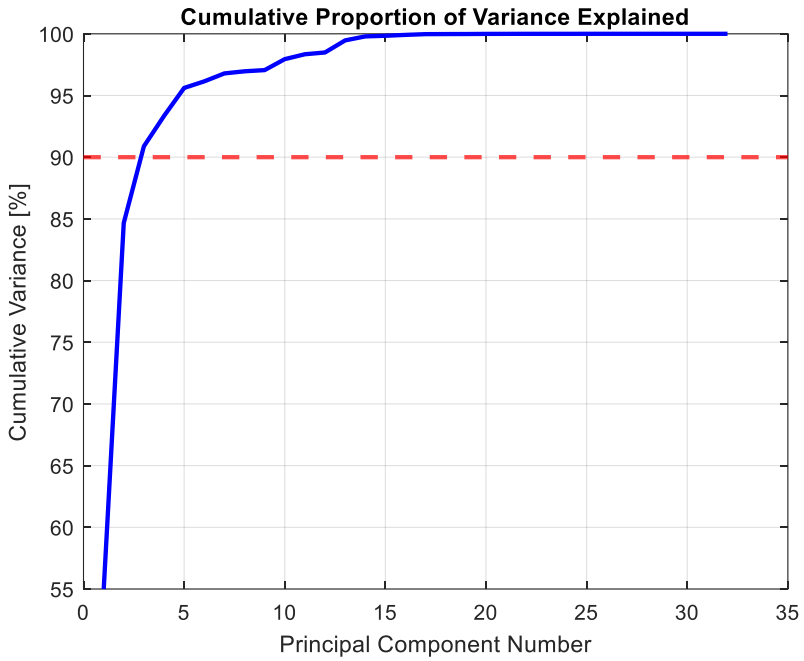

Figure 5.8: Cumulative explained variance and number of principal components. The red marker indicates the selected 90% threshold

This transformation yields a compact and decorrelated input matrix, improving both interpretability and computational efficiency. In particular, the reduction from 32 to 3 dimensions decreases the model complexity and helps to mitigate overfitting risks in subsequent regression tasks.

From a computational standpoint, the analysis was executed on an Intel® Core™ i7-6500U CPU @ 2.50 GHz. Under this configuration, the training time of the SVR model using 3 principal components was approximately 140 seconds. Although increasing the number of retained components can potentially improve prediction accuracy, it results in higher computational costs during the training phase. Therefore, a trade-off between performance and efficiency must be considered.

It is important to underline that the objective of this work is not to maximise prediction accuracy, but to develop a generalised and interpretable model capable of characterising the behaviour of a wind turbine under normal operating conditions. Such a model can then be transferred to other turbines for anomaly detection and diagnostic purposes, enhancing the robustness of the fault detection framework across different systems.

5.2.2 SVR Results: Training, Validation and Test

The SVR model was initially trained on the healthy wind turbine WT53 (see Table 4.1), using a time series of approximately 11 months (see Table 4.2). The input matrix was the 3-dimensional reduced dataset obtained from the PCA transformation, as described in Section 5.2.1. The SVR hyperparameters, such as the solver type, regularisation constant C, kernel scale, and the width of the ε-insensitive tube, were tuned to enhance the generalisation capability of the model (see Table 4.3).
In Fig. 5.9 two plots are shown based on the SVR results obtained during the training phase. Specifically, the first plot illustrates the model's predicted output, i.e. longitudinal nacelle vibration, as a function of the measured data points, while the second is the QQ-plot of the residuals.
The QQ-plot is particular important, as it provides a direct graphical representation of whether certain non linearities have not been adequately captured by the SVR. As a consequence, upper and lower bounds are computed based on the maximum perpendicular distance between the residual distribution and the reference normal distribution of the training set.

$$\kappa = r \pm \max(d)$$

Where r is the residual distribution plot into the QQ-plot, and $d = |Q - r|/\sqrt{2}$. These limits are particularly considered during the validation phase of the SVR model, where unseen data are used to assess its generalisation capability.

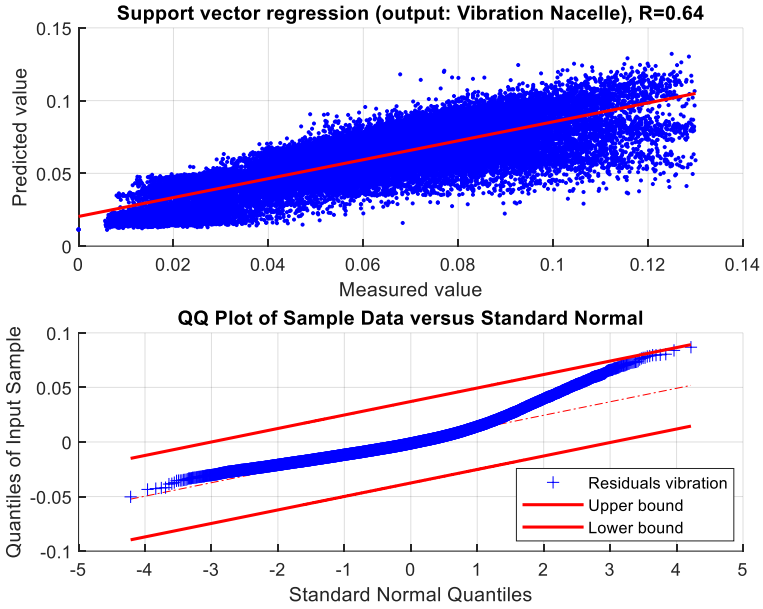

Figure 5.9: Top: predicted vs measured nacelle vibration. Bottom: residual distribution confirming near-normal behaviour

Fig. 5.10 presents the time-domain comparison between the predicted output and actual vibration data, demonstrating good agreement during the training and validation phases. The second subplot shows the residuals over time, highlighting the presence of a non-zero error due to model limitations and the inherent variability in the real-world data.

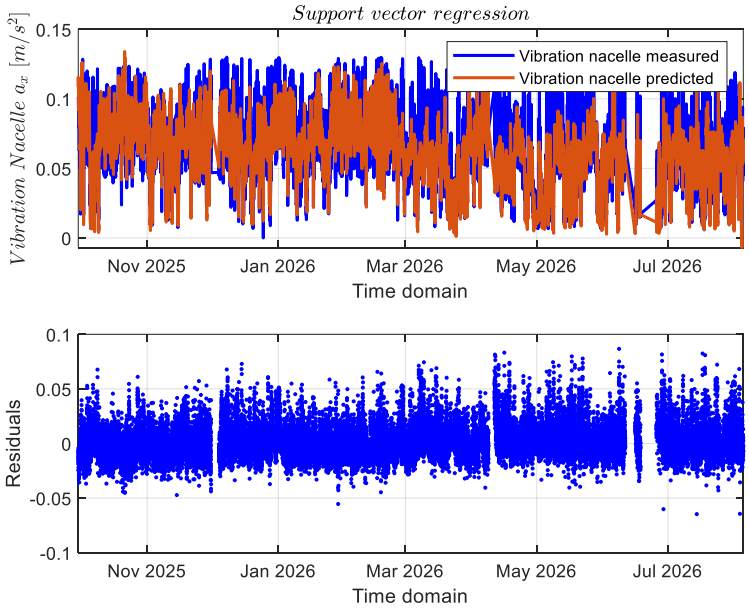

Figure 5.10: Top: time-domain predicted and measured vibration values. Bottom: residuals over time. The missing data are caused by the application of label filtering

The model was tuned iteratively by comparing training and validation performances, primarily based on the metrics reported in Table 5.4. Notably, some residuals are undefined, resulting from missing or discarded points in the dataset due to label filtering. As discussed in Chapter 3, only normal operating conditions (label 0 links to a healthy status of the turbine) were retained during training and validation, leading to the exclusion of some SCADA data labelled as fault conditions.

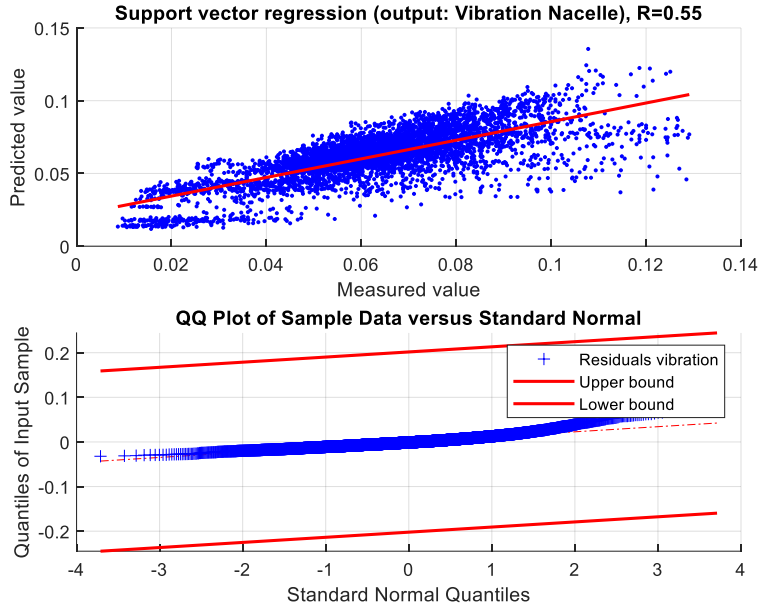

Figure 5.11: Validation results. Top Plot: Predicted value and Measured value representation. Bottom Plot: Residuals QQ-plot

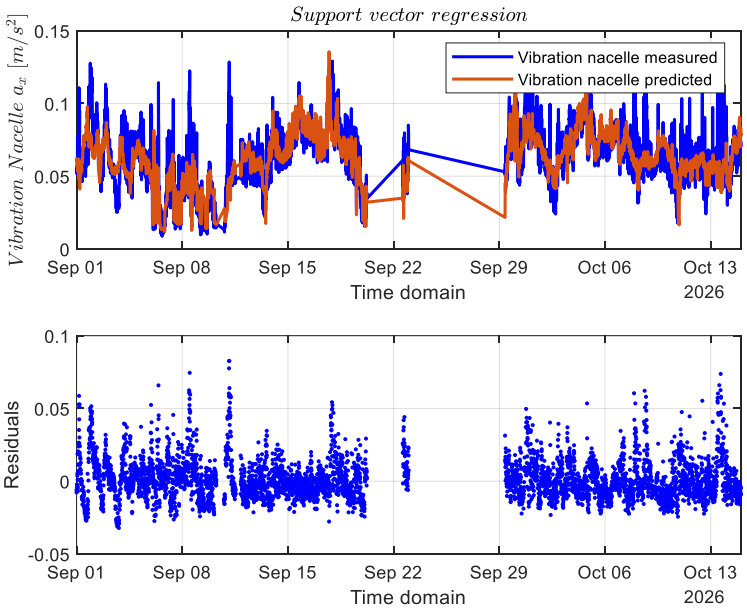

Figure 5.12: Validation results. Top Plot: Predicted and measured values plotted in the time domain. Bottom Plot: Residuals time series plot. Missing data are caused by the application of label filtering

Table 5.4: Performance metrics results concerning training and validation phases

Use	R^2 [%]	$RMSE$ [%]	MAE [%]
Training	64.1	1.5	1.1
Validation	55.4	1.4	1.0

A notable difference highlighted in Table 5.4 concerns the reduction of the coefficient of determination R^2 between the training and validation phases. This index, defined in Equation (4.28), is strongly affected by the behaviour of the residuals, as it quantifies the proportion of variance in the output that is explained by the model. Therefore, any degradation in model's predictive capability is directly reflected in a decrease in R^2, due to its inverse proportionality to the residual variance.

Specifically, an increase in the magnitude of the residuals leads to a decrease in R^2, indicating a diminished ability of the regressor to capture the underlying structure of the output signal. Ideally, $R^2 = 1$ represents a perfect fit between the predicted and actual outputs, implying that the model residuals are null. However, such an ideal condition typically corresponds to a situation where the model has learned the training data too precisely, possibly even memorising noise or irrelevant fluctuations, making it incapable of generalising accurately to new, unseen data. For this reason, this work aims to identify an optimal trade-off ensures both a satisfactory fit to the training data and robust generalisation capability.

A highly complex model is capable of following minimal variation within the training dataset, and if such variations are generated by stochastic noise or uncontrolled disturbances, the model tends to incorporate this noise into its structure. As a result, it becomes unable to generalise when exposed to new data, leading to significantly larger residuals and thus poor performance in real-world applications. Due to the inherent presence of noise and, more generally, nonlinearities affecting the physical system, it is practically impossible for any model to produce a zero-residual fit across all operational instances. Therefore, a certain level of systematic error must be expected and tolerated.

The selection of model complexity is thus a crucial aspect to control in order to minimise the residuals without overfitting. If the model is too simple, it will fail to capture essential dynamics, leading to underfitting. Conversely, a model that is too complex may achieve artificially low residuals during training but perform poorly under operational variability, even when the system is fault-free, this is known as overfitting. As a consequence, the goal in this preliminary phase is to identify a condition of optimal model complexity, i.e. a compromise between underfitting and overfitting, that enables accurate yet generalisable predictions.

At this stage, it becomes necessary to test the model on a dataset corresponding to a wind turbine that exhibits a fault condition. As specified in Chapter 3, the faulty data are

characterised by a high quality description provided in the datasets downloaded from Zenodo. Each dataset includes specific labels (as defined by the Status-ID, see Table 3.2) that clearly distinguish between normal and anomalous operating conditions. In particular, the test phase considers turbine WT35, which is affected by a pitch failure fault. From the statistical analysis of the available signals, it has been observed that turbulence can play a significant role in amplifying nacelle vibrations. This is particularly relevant in the context of pitch failure, where the pitch motor loses its ability to rotate the blades toward the optimal angle. Such failure compromises the aerodynamic regulation mechanism, preventing the turbine from aligning the blades effectively to capture the maximum available kinetic energy from the wind. The result is not only a reduction in power conversion efficiency, but also a higher dispersion of unutilised energy, which excites the nacelle structure and intensifies its dynamic response.
The results of this test are illustrated in Fig. 5.12 and 5.13, while the corresponding performance metrics are reported in Table 5.5.

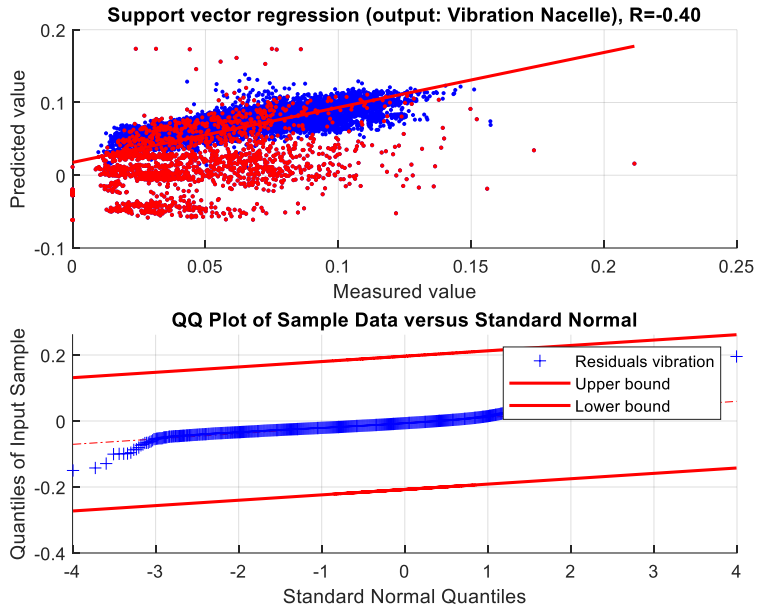

Figure 5.13: Test results. The first plot shows the predicted vibration values against the measured ones. Red markers are used to highlight data points associated with fault conditions

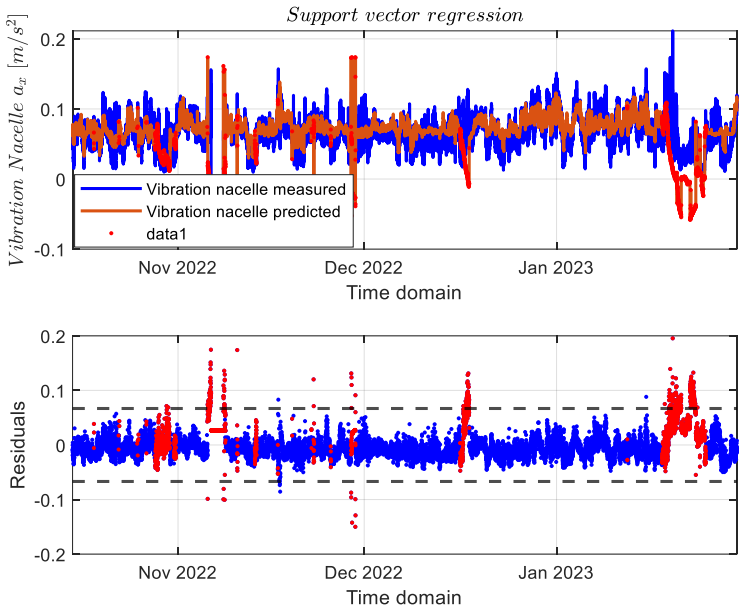

Figure 5.14: Test results. Time-domain representation of predicted, measured (top), and residual values (bottom). Faulty intervals are highlighted in red across different time windows

Table 5.5: Performance metrics: Test phase results

Use	R^2 [%]	$RMSE$ [%]	MAE [%]
Test	-40.3	2.3	1.7

From the results shown in Table 5.5, it can be observed that although both RMSE and MAE remain relatively low, suggesting that the average prediction error is not excessively large, the coefficient of determination R^2 takes a negative value. This indicates that the model's prediction during the test phase is less accurate than a naive model predicting the mean of the training outputs. In practical terms, a negative R^2 value reveals that the model fails to generalise under the altered operational conditions, and its predictions are significantly affected by error when the system deviates from the healthy regime it was trained on.

The central objective at this point is to determine whether such behaviour is indicative of a genuine failure condition or simply a consequence of insufficient model accuracy. To this end, the methodology adopted involves the definition of both lower and upper residual thresholds, derived from the statistical distribution of residuals observed during the training phase. This concept is further illustrated in Fig. 5.14.

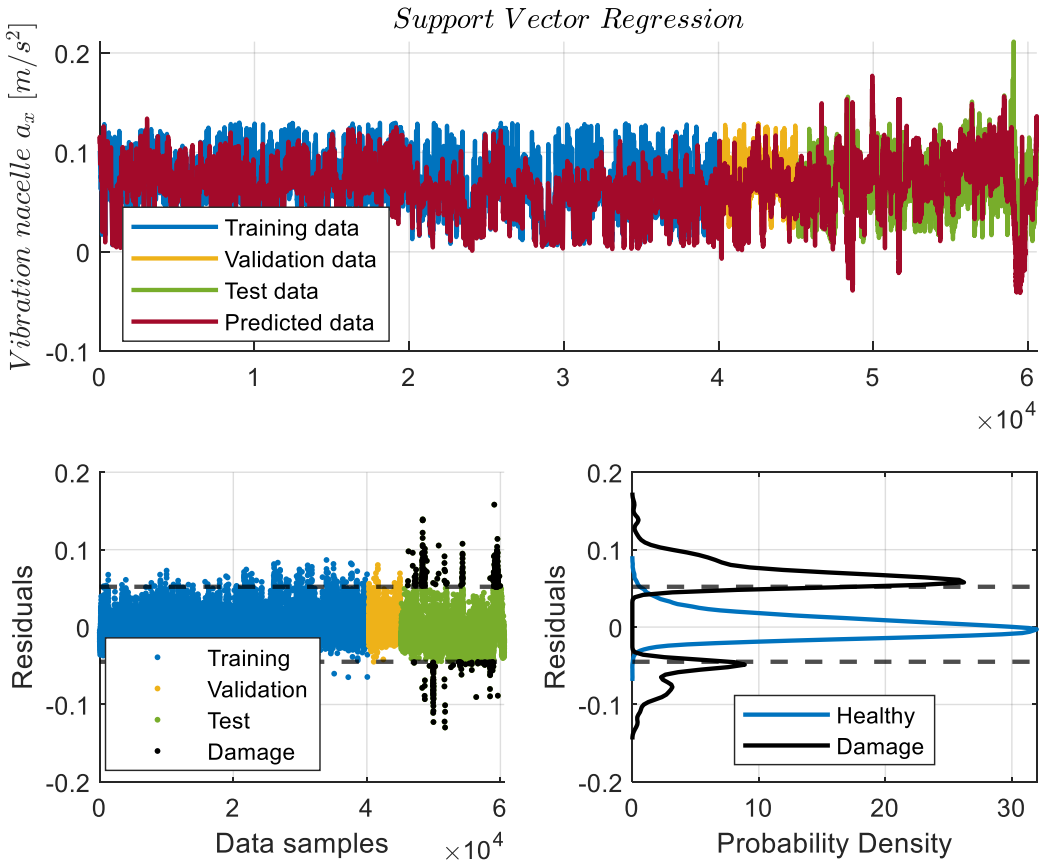

Figure 5.15: The first plot compares the predicted output with the measured outputs across training, validation, and test datasets, where the predicted values are shown in red. The second plot represents the residuals computed in the time domain, while the third plot the statistical distribution of these residuals. Residuals exceeding predefined thresholds, derived from the training distribution, are highlighted in black and classified as potential damaged points.

The analysis of the results presented in Fig. 5.14 reveals a distinct separation between the distributions of residuals associated with normal and potentially faulty operating conditions. The blue distribution corresponds to the residuals computed during the training phase, which were derived exclusively from healthy turbine states (label 0). This distribution serves as a statistical reference for what is expected under nominal behaviour. Conversely, the black-marked residuals correspond to test data points that exceed the upper or lower thresholds defined by the statistical limits of the blue training distribution. These points are flagged as anomalous because their prediction errors fall outside the confidence interval established during model training. The separation between these two distributions suggests that the model, trained only on healthy data, is capable of recognising deviations indicative of abnormal system dynamics, even in the absence of explicit fault labels during training.

This observation forms the basis for the residual-based anomaly detection strategy adopted in this work. The method assumes that, under healthy operating conditions, the

model's prediction errors (residuals) remain confined within statistically defined bounds. Therefore, any significant deviation beyond these thresholds is interpreted as a possible symptom of a structural anomaly or incipient failure in the system, such as the pitch failure considered in this study.
The validity and robustness of this detection framework, including the choice of thresholds, the false positive/false negative trade-off, and the relationship between residual distribution shifts and physical degradation phenomena, will be further examined in the following paragraph. There, a systematic evaluation of the anomaly detection strategy is conducted, aiming to quantify its performance and to confirm the model's ability to detect early-stage faults based solely on residual analysis.

5.2.3 Performance Evaluation of Univariate-Based Anomaly Detection

This paragraph presents the final results related to the anomaly detection strategy developed in the initial phase of this work, based on a univariate residual analysis. Specifically, the threshold selected from the empirical residual distribution, obtained during the training phase under healthy operating conditions, enables a manual but effective classification of normal data points in contrast to damaged ones. The purpose of this approach is to verify whether the model is intrinsically capable of learning this distinction solely through the analysis of residuals. At this stage, the availability of labelled data in the test set plays a crucial role in validating the effectiveness of the proposed method.

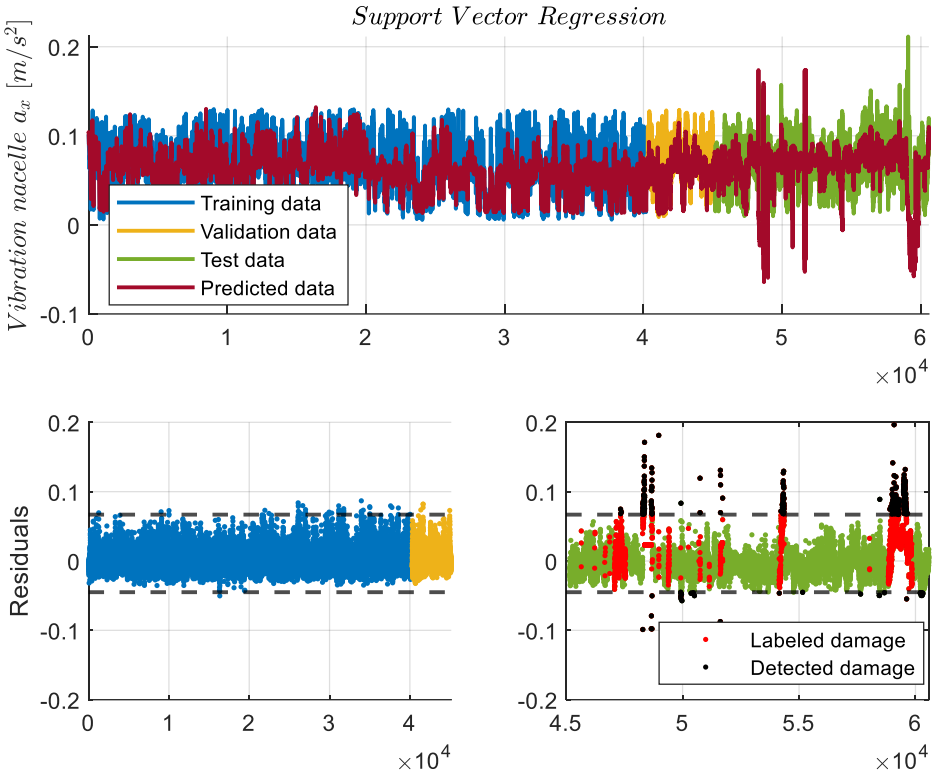

Figure 5.16: The second and third plots represent the residuals computed from the regression model in the time domain. Red dots indicate the true (labelled) damage points, while black dots correspond to the detected anomalies based on residual thresholding

As shown in Fig. 5.15, the points labelled as damaged, corresponding to a pitch-failure event, are compared with those identified by the model through the residual-based anomaly detection strategy. It is evident that the SVR model, enhanced by the use of a Gaussian kernel, is able to capture the underlying nonlinear behaviour of the system and distinguish between healthy and faulty conditions, even though the model has never been trained with faulty data. This provides clear evidence that SVR can serve as an effective tool within model-based anomaly detection framework, especially for predictive maintenance applications in wind energy systems. The major advantage lies in its capacity to monitor the health state of a turbine without requiring interruptions in power production, thereby supporting continuous operation and enabling early intervention in the case of incipient failures.

To quantitatively validate the residual-based detection method, a series of performance metrics were computed. This enables a direct comparison with other approaches found in the current state of the art. These metrics, introduced in Chapter 4, are derived from the confusion matrix and include accuracy, precision, recall, specificity, F-score, and the rates of false and missed alarms. The corresponding values are reported in Table 5.6, while Fig. 5.16 illustrates the ROC-curve associated with the variation of the decision threshold.

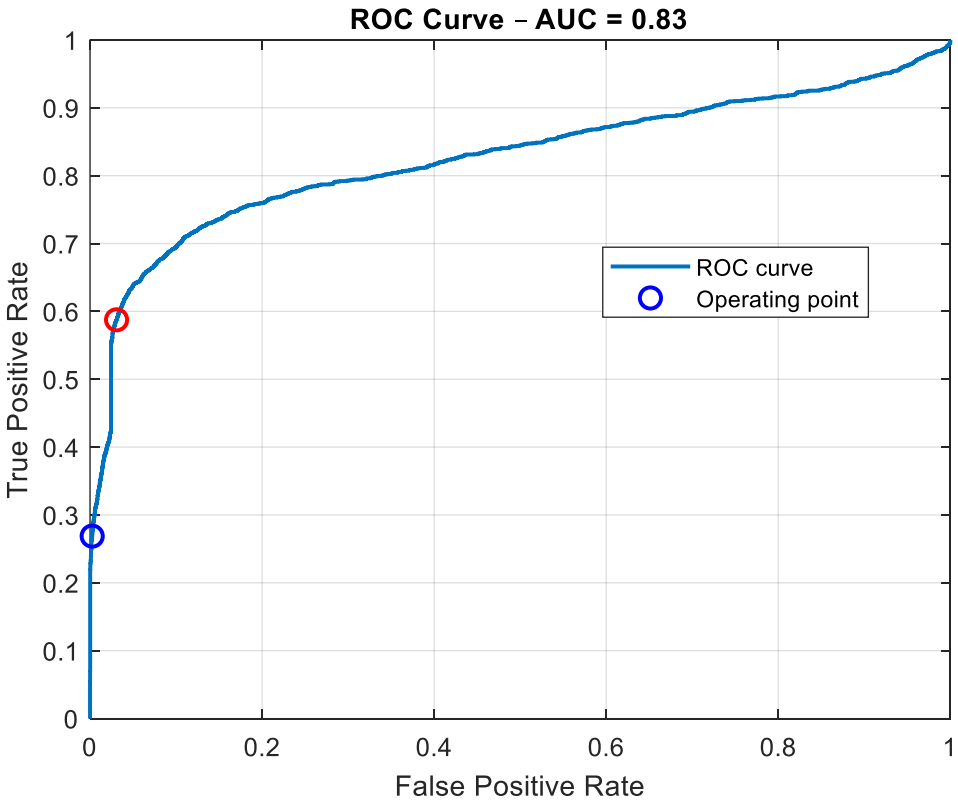

Figure 5.17: ROC curve. The red circle indicates the optimal operating point selected based on Youden's index (the "knee" of the curve). The blue circle represents the threshold defined from the residual distribution of the training set

Table 5.6: Performance metrics for different threshold values applied to residuals

Threshold	Accuracy [%]	Missed Alarms [%]	False Alarms [%]	Precision	Specificity	Recall	F-score
Blue Circle	88.76	10.92	0.32	0.91	0.89	0.24	0.26
Red Circle	83.55	5.12	11.32	0.45	0.94	0.64	0.62

As shown in Table 5.6, both thresholding strategies yield high overall accuracy, exceeding 83%. The "blue circle" threshold, derived from the training residual distribution, prioritises minimising false alarms (FA), a relevant objective in contexts where false positives incur operational or economic costs. In contrast, the "red circle" threshold, chosen at the knee of the ROC curve via Youden's index, seeks to balance false alarms (FA) and missed alarms (MA), achieving a significantly higher recall (64%), but at the cost of increased FA (11.32%).

In the context of wind turbine condition monitoring, minimising missed alarms is often more critical than minimising false alarms. A missed detection of a developing fault can lead to serious mechanical failures, unplanned shutdowns, increased downtime, and

elevated maintenance costs. For this reason, even though the blue-circle threshold yields the best performance in terms of specificity and low FA, the red-circle threshold is preferred, as it ensures a lower rate of missed faults (MA<5%), while maintaining FA below 15%.

A further insight comes from the F-score, which is the harmonic mean of recall and precision. In this work, a custom β-wieghted version of the F-score is adopted to emphasise recall, since missing an anomaly in a critical system like a wind turbine poses a significant risk. The best compromise is again achieved at the red-circle point, where the F-score reaches 0.62, indicating a moderate but promising detection capability.

Nonetheless, the performance is not yet optimal. The current detection approach, while functional, still leaves room for improvement, particularly in enhancing the separability of fault vs healthy patterns and reducing false positives without sacrificing recall. The next paragraph explores strategies to improve these aspects through more advanced or multivariate models.

In addition to the SVR model trained to predict nacelle vibrations, a second SVR model was developed in this work with the objective of predicting the active power output produced by the wind turbine. The rationale behind this attempt was to investigate whether power data, being an essential performance indicator, could serve as an additional anomaly detection metric. However, several critical issues emerged, primarily related to the high sensitivity of power output to internal and external fluctuations, as highlighted in Fig. 5.17.

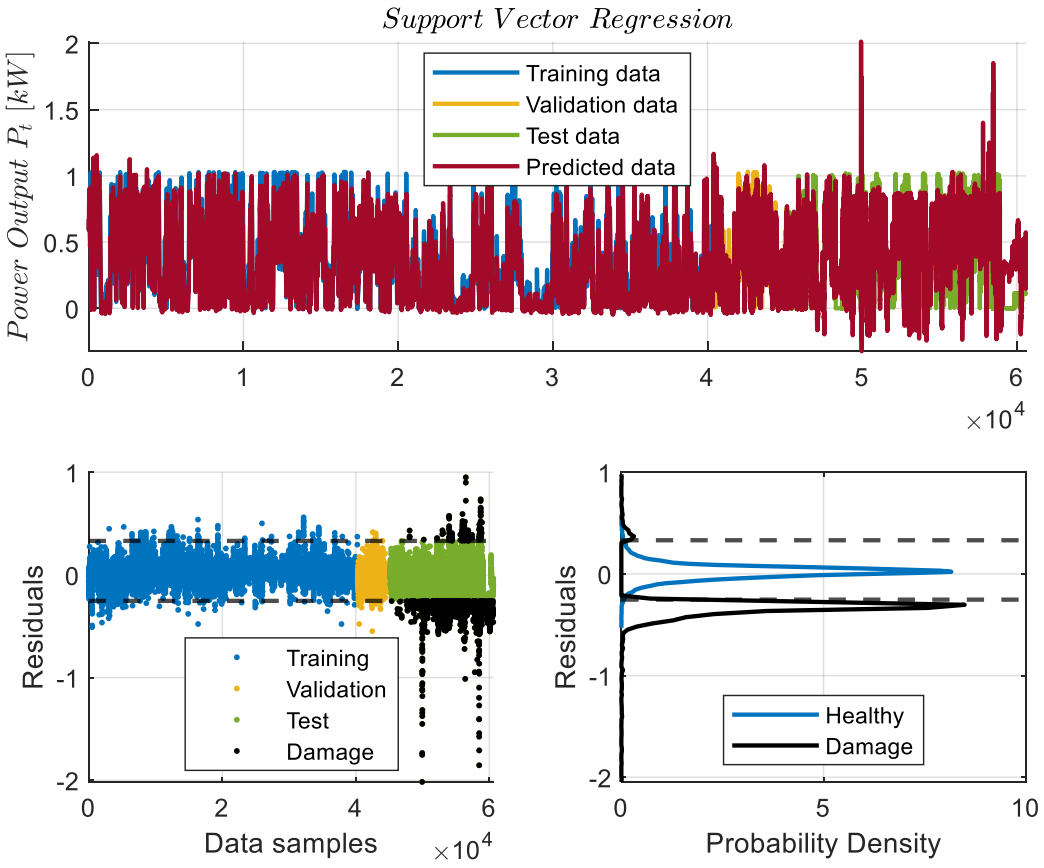

Figure 5.18: Predicted power output obtained using a secondary SVR model. In the bottom plot, black dots are referred to anomalies classified by the model, for data points that exceed the upper and lower thresholds

As observed in Fig. 5.17, the model exhibits larger prediction errors, with residuals distribute across a wide range of values, even during periods classified as healthy by the reference labels. This residual behaviour is not correlated with the fault labels and does not follow a distinguishable pattern associated with the known pitch-failure condition. As a consequence, the corresponding ROC-curve (Fig. 5.18) reveals a low AUC, indicating that the model is unable to reliably separate normal from faulty operating states.

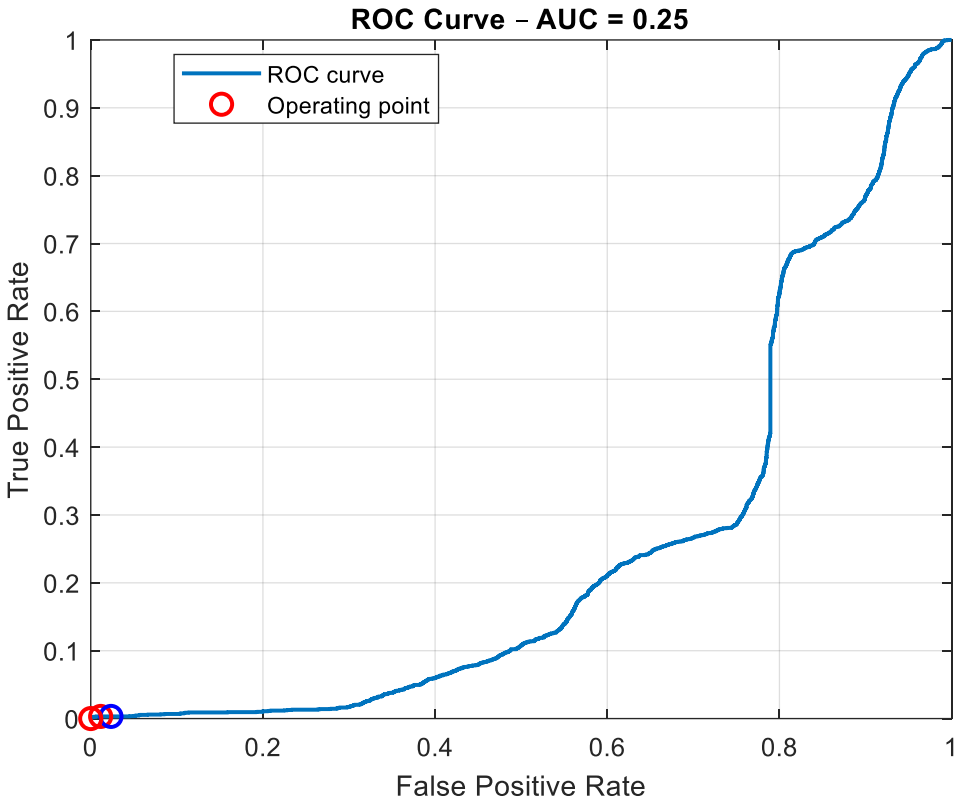

Figure 5.19: ROC-curve obtained from the SVR model trained on power output residuals

This experimental attempt was aimed at exploring whether the residuals of power output prediction could function as an effective anomaly indicator. However, power and efficiency metrics are inherently difficult to model with high reliability due to the intrinsic complexity of wind turbine systems. Even small deviations from the nominal operating conditions, originating from mechanical, electrical, or aerodynamic subsystems, can lead to abrupt and nonlinear variations in power generation. Moreover, such deviations may not necessarily correspond to a physical fault, but instead reflect momentary disturbances, control system interventions, or grid-related constraints.

In particular, the SCADA system tends to record zero power output when a system-wide error is detected, such as protective shutdown or trip condition. These zero values introduce discontinuities and further degrade the accuracy of the regression model. As a result, the model frequently identifies false anomalies, i.e. operating conditions flagged as faulty by the residuals, despite the turbine functioning correctly.

The quantitative performance of this second model is summarised in Table 5.7. The AUC value of 0.25 confirms that the model performs worse than random guessing, and the false alarm rate exceeds 25%, as seen from the blue-circle threshold, rendering this approach unsuitable for effective anomaly detection in its current form.

Table 5.7: Performance metrics of the anomaly detection model

Threshold	Accuracy [%]	Missed Alarms [%]	False Alarms [%]
Blue Circle	71.1	3.4	25.1

Given these limitations, and the risk of introducing a high number of false positives, it is concluded that power output is not a reliable variable for residual-based anomaly detection within the proposed framework. Therefore, in the remainder of the analysis, only vibration signals will be used as the primary monitoring variable for fault detection purposes, due to their higher sensitivity and correlation with structural anomalies.

5.3 Linear Regression Power-Wind Speed

Before proceeding with the multivariate analysis, as introduced in Fig. 4.1 and motivated by the limitations discussed in Chapter 4, a complementary analysis has been carried out alongside the assumptions based on the Betz model coefficients. This analysis is part of the broader preprocessing strategy, and its primary objective is to model, in a simplified yet informative manner, the relationship between wind speed and generated power, with the aim of refining both regression accuracy and anomaly detection effectiveness.

The motivation behind this approach stems from the lack of detailed design information about the wind turbine under investigation. In fact, no direct information is available regarding its rotor diameter, efficiency, or power coefficient C_p, all of which would be required for a physics-based model. As a result, all functional insights have been inferred from statistical analysis of SCADA data, an approach that is inherently subject to systematic measurement uncertainties and limitations related to operational variability.

Instead of assuming specific geometric parameters or ideal performance coefficients, a simple linear regression model, based on a third order polynomial function, has been developed to estimate the slope of the power-wind speed curve, as outlined in Equation (4.14). This estimated slope, denoted as $\{\beta_i\}$, is then intended to be included as a new feature within the input matrix, where each term is multiplied by the corresponding power of wind speed, up to the third order, thereby contributing to the improvement of the final multivariate prediction model. The polynomial order was selected by considering the main dependencies described by the Betz formula, which relates power output to wind speed.

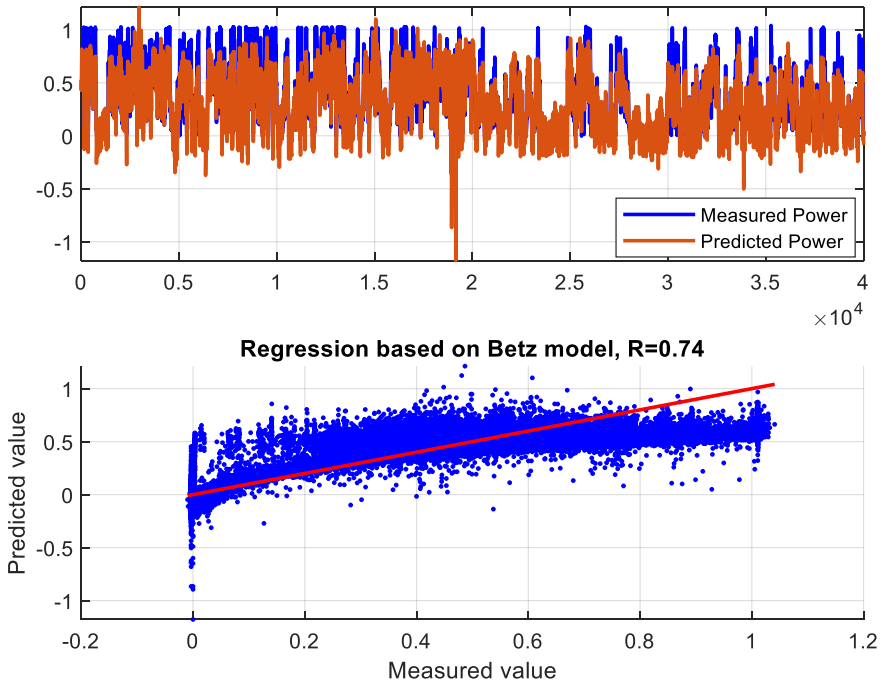

Figure 5.20: Comparison between predicted and measured power output responses in the time domain using the linear regression model

From the results illustrated in Fig. 5.19, it is evident that the predicted power output matches the measured values reasonably well, suggesting a satisfactory performance of the linear model. This is further supported by the coefficient of determination, with a value of $R^2 = 0.74$, indicating that the model is capable of explaining approximately 74% of the variance in the observed data. Although this does not reflect a perfect fit, it confirms that the cubic formulation is sufficiently adaptable to the actual data to be used for further anomaly-related investigations.

The detailed results for all three phases (training, validation, and test) are reported in Table 5.8.

Table 5.8: Performance metrics of the linear regression model using the Betz-based formulation

Use	$\boldsymbol{R^2}$ [%]	***RMSE*** [%]	***MAE*** [%]
Training	64.5	1.5	1.1
Validation	58.8	1.4	0.9
Test	-63.9	2.3	1.8

From these metrics, one can observe a consistent performance between the training and validation phases, with R^2 values exceeding 50% and very low error percentages.

However, a notable drop in performance is observed during the test phase, where the coefficient of determination becomes negative. This reflects the model's inability to generalise to unseen data when fault conditions are present, which, paradoxically, becomes an advantage in anomaly detection.
Indeed, the increase in residuals during faulty conditions enhances the distinguishability between normal and abnormal states, which is beneficial for a residual-based detection framework. The corresponding ROC curve shown in Fig. 5.20 confirms this behaviour.

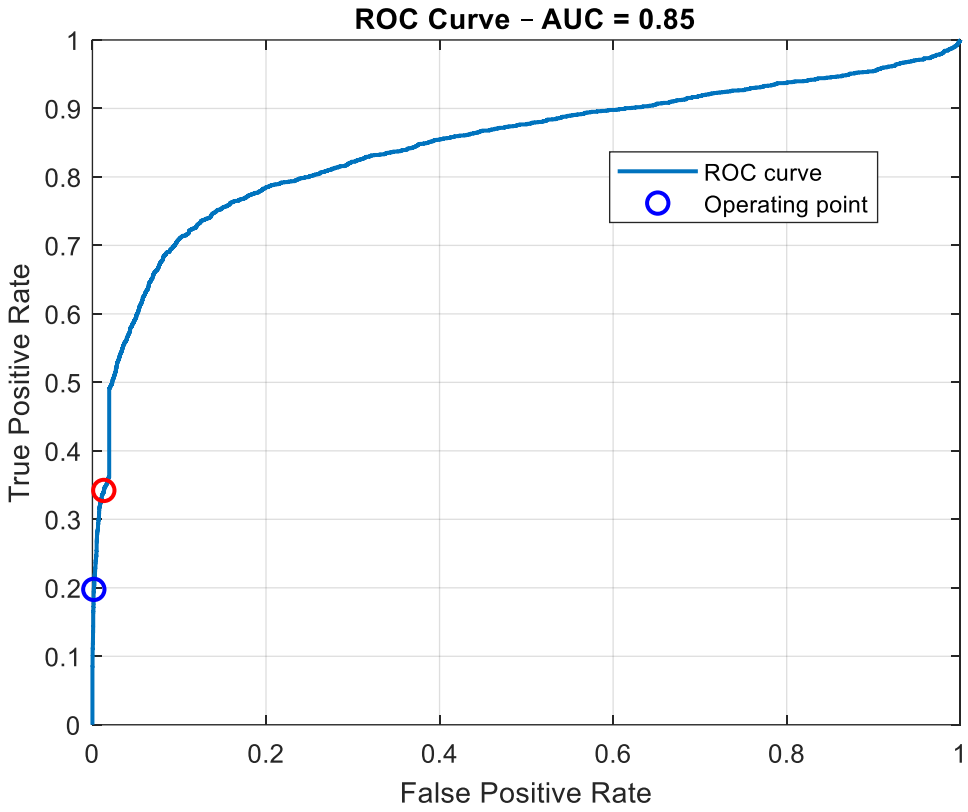

Figure 5.21: ROC-curve representation for the linear regression model. The red marker indicates the optimal threshold, while the blue marker represents the threshold derived from the residual distribution during training

The separation between false alarms and true positives along the ROC curve validates the hypothesis that even a simple linear model, if sufficiently correlated to physical behaviour, can provide a valuable indicator for fault detection purposes, especially when detailed turbine-specific information is not available.

Table 5.9: Performance metrics of the anomaly detection model based on third-degree polynomial regression

Threshold	Accuracy [%]	Missed Alarms [%]	False Alarms [%]	Precision	Specificity	Recall	F-score
Blue Circle	87.3	12.1	0.7	0.8	0.9	0.3	0.2
Red Circle	82.2	8.9	8.9	0.4	0.9	0.4	0.4

From the analysis of Table 5.9, it can observe that this approach yields moderate classification performance in terms of anomaly detection. Although the ROC curve shows an acceptable separation between classes, the F-score values are significantly lower than those obtained in the previous univariate analysis where the regression was derived directly from the assumed Betz formulation. Specifically, even when adjusting the operating threshold to balance the trade-off between missed alarms and false alarms, the maximum F-score does not exceed 0.38, indicating a limited capability to simultaneously achieve both high sensitivity and precision.

This performance gap can be interpreted as a consequence of the regression model's reduced physical interpretability and the increased variability introduced by directly estimating the coefficients from noisy SCADA data, without enforcing physical constraints.in contrast, the Betz-inspired model, although idealised, provides a more stable baseline for univariate anomaly detection, as it encodes prior aerodynamic knowledge and inherently respects the cubic law of power extraction.

As a result, it can be concluded that, for the specific case of univariate analysis, the direct application of Betz's law yields more robust and reliable anomaly performances.

Nevertheless, these findings also reinforce the notion that univariate models alone are insufficient to fully capture the complexity of fault-related dynamics in wind turbines. Therefore, in the following paragraph, a multivariate analysis is proposed, incorporating multiple output variables and capturing higher-dimensional interactions among features. This step is intended to improve the model's generalisation capability, enhance sensitivity to early fault signatures, and reduce the limitations encountered with single-variable approaches.

5.4 Multivariate Analysis

The multivariate analysis has been conducted by training four distinct SVR models, each developed to learn the healthy operational behaviour of wind turbine WT53. Each model is designed to predict one of the four vibration signals acquired from the accelerometers mounted on the nacelle. As introduced in Fig. 4.10, the objective at this stage is to define a new anomaly index, different from that used in the univariate analysis. In particular, residuals obtained from the individual SVR regressors are aggregated into a single multivariate residual vector, which is then used as a comprehensive anomaly indicator. Two distinct approaches are adopted to evaluate this index: Mahalanobis distance and One-class SVM. These methods follow the same underlying principle used in the univariate approach: any observation whose index exceeds a defined threshold is classified as anomalous.

The motivation behind the dual-model strategy lies in their different mathematical foundations and operational sensitivities, offering complementary advantages in anomaly detection. As in the univariate case, the input matrices were constructed in two alternative ways: by incorporating the theoretical coefficients derived from the Betz model, and by including the empirically derived $\{\beta_i\}$ coefficients from the polynomial regression of the power-wind speed relationship (as described in the preprocessing section).

Fig. 5.21 shows the anomaly index computed via the Mahalanobis distance across training, validation, and test datasets. The true failure period of the damaged turbine WT35 is highlighted in red. Visually, a clear separation is observable between healthy and faulty data points, although some isolated exceedances of the threshold occur even during the training phase. These may be attributed to systematic noise or data uncertainties inherently present in SCADA systems.

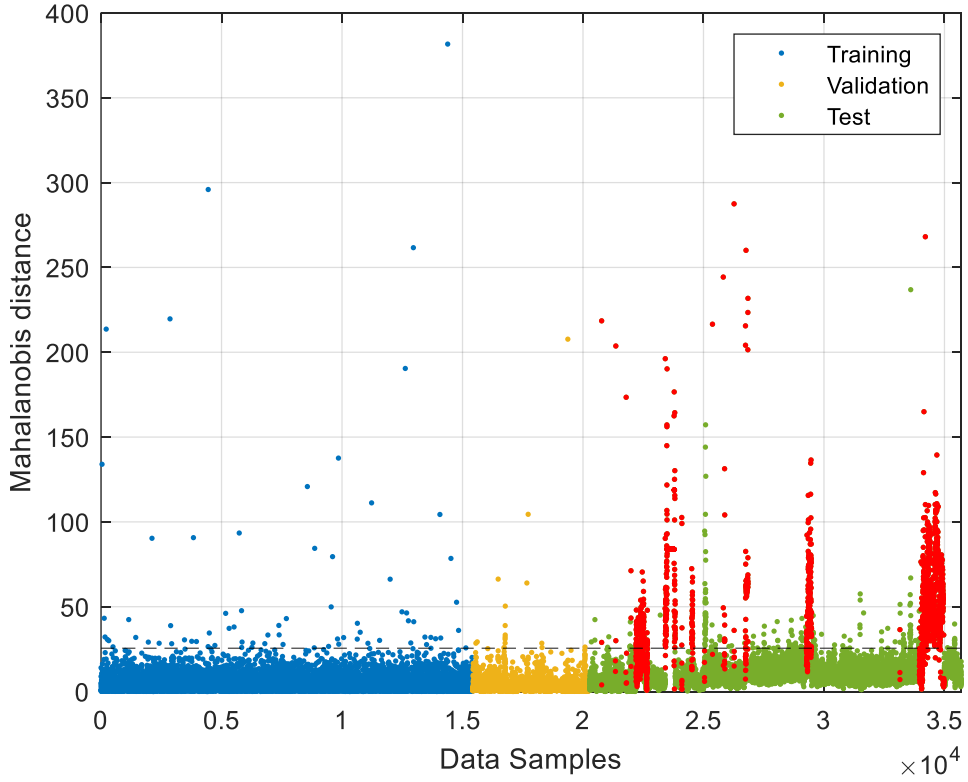

Figure 5.22: Mahalanobis distance plot. Red markers denote the labelled fault data corresponding to the true failure period of WT35

The multivariate analysis, derived from the four-sampling vibration signal (as discussed in Chapter 4), provides a richer representation of the system's dynamic behaviour. This is particularly beneficial when computing the Mahalanobis distance, which not only accounts for the magnitude of the residuals but also incorporates their covariance structure. Specifically, as shown in Equation (4.32), Mahalanobis distance is computed starting from the covariance matrix of the healthy data points, and this approach enhances sensitivity to correlated anomalies in multiple signals. From a physical standpoint, increasing the number of vibration signals (i.e., using multiple accelerometers positioned on the nacelle) enables the detection of a wider range of vibrational modes and localised structural responses, improving the model's fault detection capability. These improvements are substantiated by the performance metrics reported in Table 5.10.

The first evaluation is based on the Betz-derived input configuration. Fig. 5.22 and 5.23 show the ROC curves corresponding to the Mahalanobis distance and One-class SVM respectively. While Mahalanobis distance is based on the computation of the covariance matrix linked to the training residuals (healthy data), also One-class SVM method is especially suitable in this context because the training set includes only healthy data. It identifies anomalies in the test data by detecting points that deviate significantly from the

training data distribution, i.e. by learning a boundary that encloses the nominal operational regime.

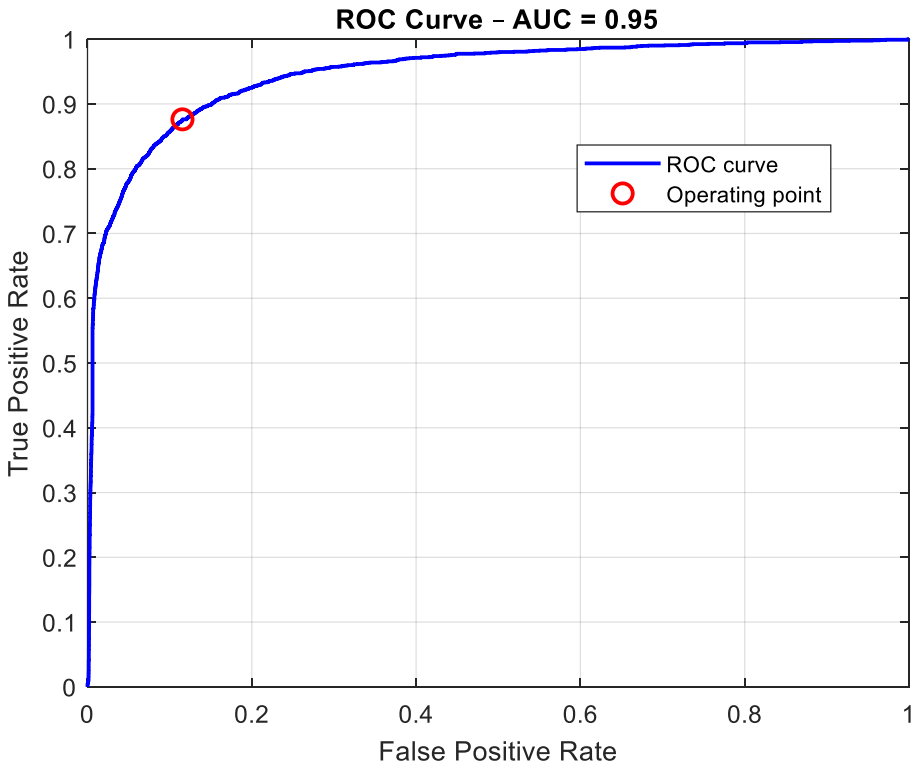

Figure 5.23: ROC curve computed from Mahalanobis distance anomaly index

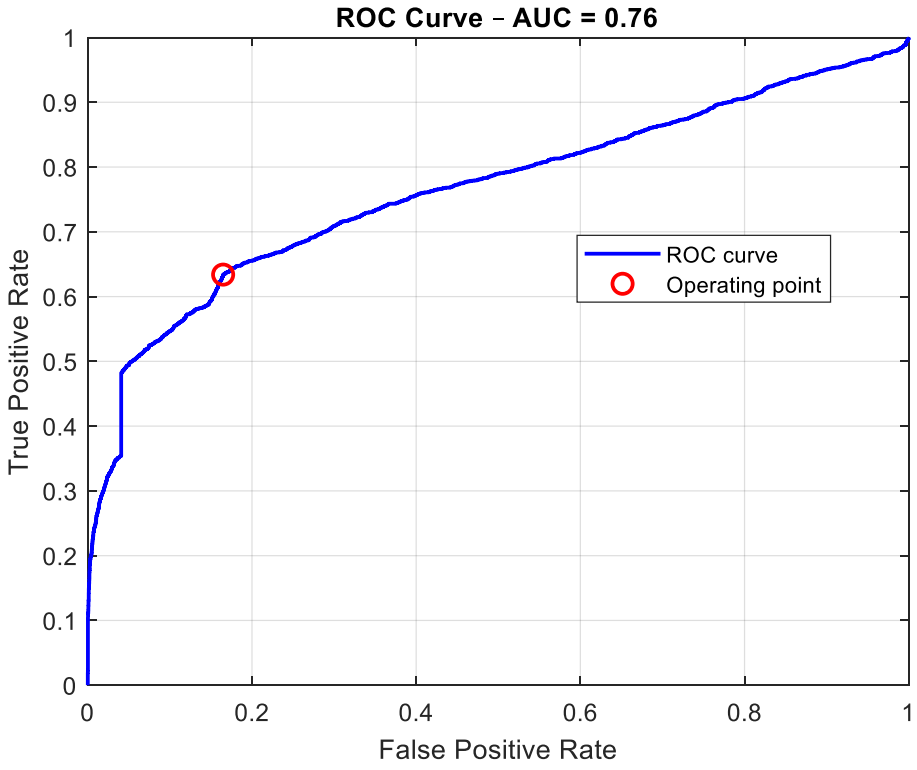

Figure 5.24: ROC curve computed from One-class SVM score

The main improvement over the univariate analysis is evident when applying the Mahalanobis distance, which achieves an AUC of 0.95. This indicates that the model possesses excellent discriminative capability, with at least a 95% probability of assigning a higher anomaly score to a positive instance than to a negative one when comparing randomly selected samples. The result obtained using the One-class SVM is slightly lower than that of the Mahalanobis distance, which may be attributed to the continuous operation of the turbine: when an anomaly arises, maintenance is typically performed immediately. This behaviour influences the dataset structure, resulting in a predominance of normal data and a limited number of faulty instances, thereby making the distinction between the two classes less pronounced. In addition, this difference is not attributable to the linearity of OCSVM, as the use of a Gaussian kernel did not leads to improved results (Fig. 5.27), and this suggest that the limitation lies in the intrinsic characteristic of the OCSVM, which defines a decision boundary that may be overly sensitive to anomalies or to the class imbalance present in the dataset.
Moreover, the One-class SVM defines a hyperplane to separate normal from anomalous data. In this context, where class overlap exists, this separation becomes more challenging. In contrast, Mahalanobis distance measures how far a point lies from the distribution of normal data, considering both the mean and the covariance structure of the training data (healthy data points). This makes it particularly suitable when classes are partially overlapping, as anomalies, though not sharply distinct, are progressively more distant from the nominal condition and thus more easily detectable.

Table 5.10: Performance metrics-Multivariate anomaly detection models

Anomaly index	**Accuracy [%]**	**Missed Alarms [%]**	**False Alarms [%]**	**Precision**	**Specificity**	**Recall**	**F-score**
Mahalanobis	88.3	1.9	9.9	0.6	1.0	0.9	0.8
One-class SVM	80.6	5.3	14.1	0.4	0.9	0.6	0.6

The improvement over the univariate model is evident, especially for Mahalanobis distance, which achieves a MA rate below 2%, while keeping FA below 10%. Furthermore, the Recall increases significantly, resulting in a notable improvement in the F-score.

Having established the performance of both Mahalanobis and One-class SVM under Betz-based inputs, it is now relevant to investigate how the regression-based feature affects model behaviour in the multivariate case.

Figs. 5.25, 5.26 and Table 5.11 report the results obtained by employing the $\{\beta_i\}$ coefficients derived from the regression model of the power-wind speed relationship.

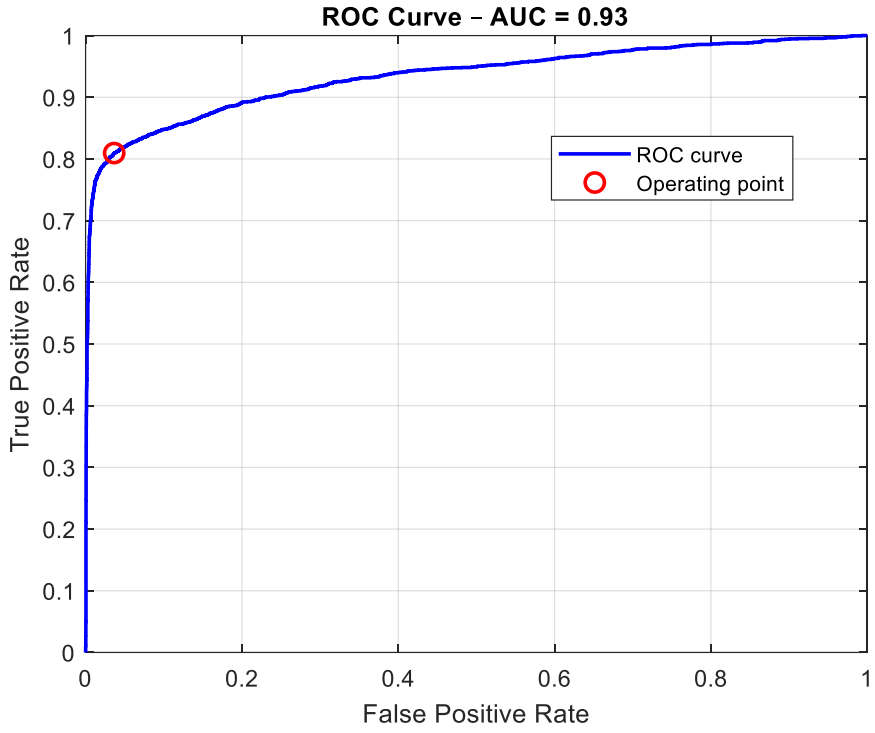

Figure 5.25: ROC-curve obtained using Mahalanobis distance based on the residuals of the Multivariate SVR model

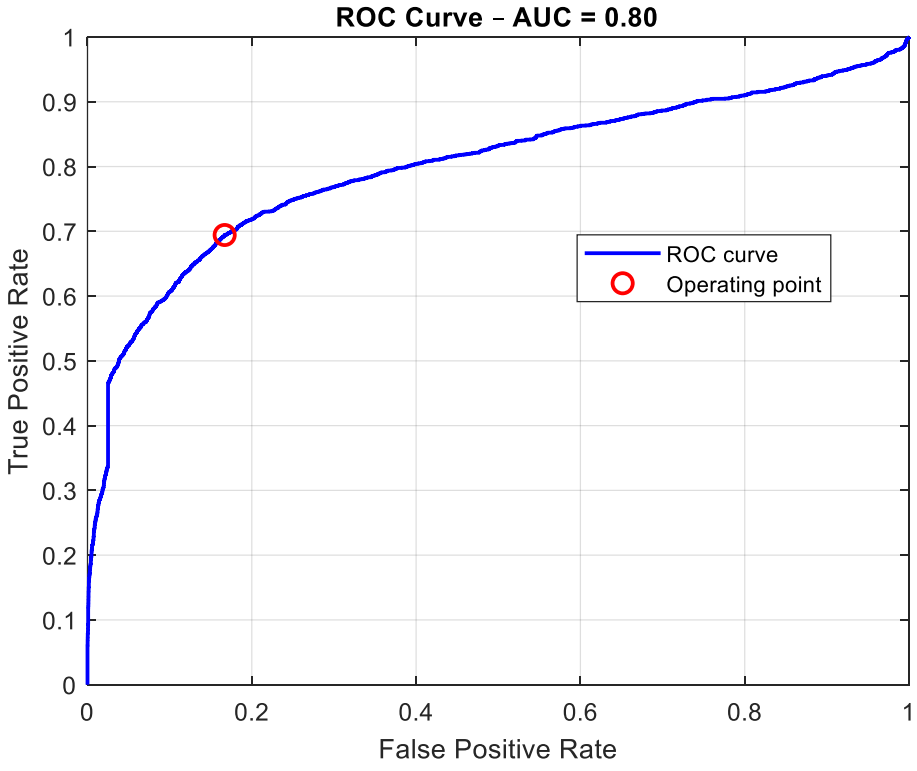

Figure 5.26: ROC-curve obtained using the OCSVM with a linear kernel

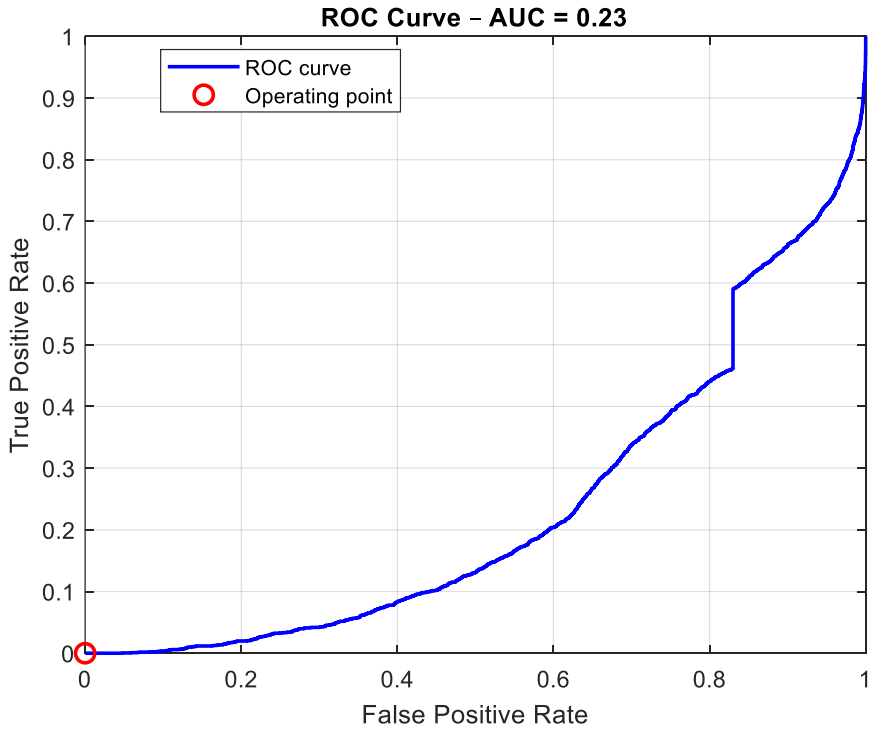

Figure 5.27: ROC-curve obtained using the OCSVM with a Gaussian kernel. The AUC value below 0.5 indicates that the model fails to correctly distinguish between normal and anomalous instances, possibly assigning higher anomaly scores to healthy data

A reduction in the AUC derived from Mahalanobis distance is observed, while an increase in the AUC for the One-class SVM is noted. Table 5.11 further highlights a key observation: both MA and FA decrease, with a concurrent improvement in Precision and Accuracy. This suggests that a more detailed preprocessing, particularly the regression-based modelling of the power-wind speed curve, enhances the model's diagnostic capability. Even the One-class SVM achieves more accurate results, highlighting the added value of including a feature that better characterises the physical behaviour of the turbine.

Interestingly, this improvement is not observed in the univariate analysis, where model performance seems to decrease. This may be explained by the fact that in the multivariate case, the inclusion of additional outputs allows for a more comprehensive representation of the nacelle's dynamic behaviour under varying load conditions. As such, the regression-based feature helps to capture variance that would otherwise remain unexplained. Conversely, in the univariate model, where only a single output is considered, the added feature may introduce noise or irrelevant information, offering limited contribution to anomaly detection.

Table 5.11: Performance metrics-Multivariate analysis results

Anomaly index	Accuracy [%]	Missed Alarms [%]	False Alarms [%]	Precision	Specificity	Recall	F-score
Mahalanobis	94.1	2.7	3.1	0.8	1.0	0.8	0.8
One-class SVM	81.3	4.4	14.3	0.4	0.9	0.7	0.7

In conclusion, the multivariate analysis yields more robust and reliable results compared to the univariate approach, mainly because it leverages the correlation structure among variables. The Mahalanobis distance, in particular, evaluates how far each data point deviates from the expected normal condition, while accounting for the interdependencies among variables through the covariance matrix. This enables the detection of subtle anomalies that may remain below threshold in a univariate context. Furthermore, multivariate analysis inherently reduces the impact of noise and confounding factors, increasing the discriminative power of the model. In a unidimensional space, normal and anomalous data may be overlapped and therefore difficult to separate, whereas in a higher-dimensional feature space, the separation becomes clearer.

With regard to the two approaches adopted during the preprocessing phase, namely, the direct application of the Betz hypothesis through its analytical formulation, and the regression model used to represent the power-wind speed relationship, a significant effect can be observed from this analysis. In the multivariate framework, the regression coefficients prove to be more informative as an additional feature when compared to the Betz-based formulation, which is derived from ideal assumptions. The regression-based feature allows for a more accurate separation between normal and anomalous operating conditions, improving the discriminative capability of the model.

In the univariate analysis, where residuals are used directly as the anomaly index, the inclusion of the regression-derived feature leads to an increased AUC, meaning that the model improves its ability to correctly rank normal and anomalous samples. Conversely, the use of the Betz formulation enables the detection of a greater number of anomalies, as evidenced by the higher Recall and the lowest MA rate, despite the cost of a higher FA rate. This outcome may be attributed to the nature of the residual distribution: the regression-based approach, being empirically driven, may introduce higher variance or noise, whereas the Betz formulation, by relying on a simplified physical law, generates residuals with smaller deviations and a more regular distribution.

This increased variance observed with the regression-based input is effectively captured by the Mahalanobis distance in the multivariate analysis, which takes into account the covariance structure of the data. As a result, it enhances the ability to distinguish between normal and anomalous states, especially when class boundaries are not sharply defined.

Therefore, while both features offer complementary advantages, the regression-based input provides greater flexibility and better integration within a multivariate diagnostic framework, especially when the objective is to maximise anomaly separability in real-world operational data.

5.5 Classification Results

The final paragraph presents the results of the classification process, developed on top of the multivariate anomaly detection framework. In this final paragraph, it will be shown the results obtained from a multi-class analysis, involving four different wind turbines, as summarised in Table 5.12.

To construct the training set for classification, a matrix was assembled by collecting the residuals obtained from the four SVR models, previously trained on the healthy operating behaviour of wind turbine WT53. Each column of the matrix corresponds to the residuals from one SVR model, and includes data from the training, validation, and test phases related to turbine WT35, which is affected by a known fault (Pitch-failure). Additionally, residuals from a third turbine (WT15), affected by a different failure, were appended to the same structure. This turbine exhibited distinct fault conditions, involving a P20 brake disc grounding role and a P20 cover lightning main cabinet hub, both of which can induce dynamic imbalances or localised resonant responses, clearly detectable through vibration measurements. The fourth turbine considered is WT52, which is affected by a carbon brush defect and an issue in the hydraulic system accumulator. Both faults can significantly increase nacelle vibrations, as they involve mechanical components located within the same housing.

In this analysis, since each wind turbine is affected by a different type of anomaly and maintenance interventions are typically carried out in a timely manner, the dataset contains more normal data than anomalous data. Furthermore, the duration of each fault varies from turbine to turbine, contributing to class imbalance. As a result, the final dataset is considered unbalanced, which may affect the performance and reliability of classification results.

The resulting dataset therefore includes four distinct operating conditions, as shown in Table 5.12.

Table 5.12: Overview wind turbines under analysis and classes identification

Turbine	Class	Status
WT53	0	Normal behaviour
WT35	1	Pitch-failure
WT15	2	P20 brake disc grounding role and P20 cover lightning main cabinet hub
WT52	3	Carbon brush defect (spinner) and hydraulic system accumulators' issue

The final residual matrix consists of four columns, each representing one SVR model and rows corresponding to the different time-series segments from each class (Fig. 5.26). This multivariate residual space serves as the input feature domain for classification.
Two supervised classification methods were applied:

- Linear Discriminant Analysis (LDA)
- Quadratic Discriminant Analysis (QDA)

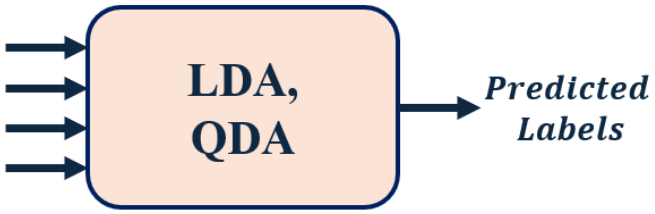

Figure 5.28: Classification scheme

To ensure a robust estimation of generalisation performance, a 10-fold cross-validation was implemented across the entire dataset (Fig. 5.27). The classification performance was then assessed using a confusion matrix, enabling the computation of class-errors and overall accuracy, as defined in Chapter 4.

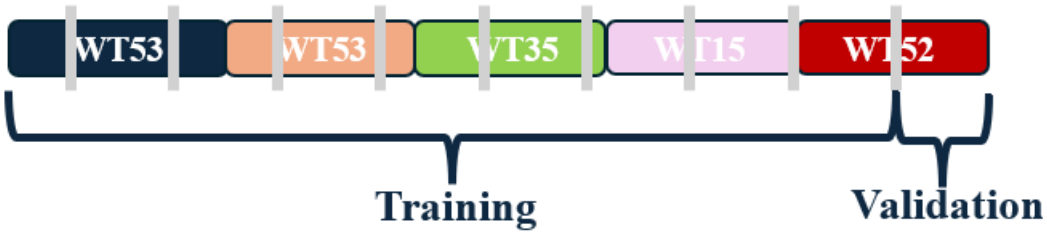

Figure 5.29: Visual representation of the 10-fold cross-validation applied to the residual dataset used for training and evaluating the classification models

The structure of the confusion matrix in this four-class scenario is reported in Table 5.13.

Table 5.13: Confusion matrix structure

	True Class 0	True Class 1	True Class 2	True Class 3
Predicted Class 0	TN_0	FN_1	FN_2	FN_3
Predicted Class 1	FP_1	TP_1	CE_{12}	CE_{13}
Predicted Class 2	FP_2	CE_{21}	TP_2	CE_{23}
Predicted Class 3	FP_3	CE_{31}	CE_{32}	TP_3

Table 5.14: Confusion matrix result starting from QDA

61894	661	448	167
446	1743	6	26
448	4	373	3
571	81	12	2601

Table 5.15: Confusion matrix result starting from LDA

62942	133	95	0
770	1431	7	13
722	15	91	0
810	14	0	2441

Table 5.16: Performance Metrics-Classification results

Classification method	**Accuracy [%]**	**MA [%]**	**FA [%]**	**Precision**	**Recall**	**F-score**	**Class-Error [%]**	**Performance Index [%]**
LDA	96.3	3.3	0.3	0.9	0.6	0.9	1.2	91.7
QDA	95.9	2.1	1.8	0.8	0.8	0.8	2.7	89.7

From the results presented in Tables 5.14, 5.15 and 5.16, it is evident that LDA shows better overall performance than QDA, achieving higher accuracy, F-score, and performance index. Despite QDA achieving a marginally lower MA, LDA exhibits a lower FA, higher precision, and a better balance between detection sensitivity and class separation. This performance difference can be attributed to the underlying assumptions of the two algorithms (see Chapter 4). LDA assumes homoscedasticity, meaning that all classes share the same covariance matrix, which leads to linear decision boundaries in the feature space. QDA, on the other hand, estimates a separate covariance matrix for each class, allowing for curved and non linear decision surfaces. While this flexibility can be beneficial when class distributions are highly non linear, it may also lead to overfitting or degraded performance in the presence of noise, limited data, or weakly separable classes. As a result, QDA is more suitable when class distributions are non-linearly separable, as in the case of real-world fault detection where failure modes can manifest in different, complex ways. Moreover, while LDA constructs hyperplanes to separate classes, QDA incorporates quadratic terms, allowing for hyper-ellipsoid, hyper-paraboloid, or even hypersphere boundaries. However, in this specific case, such flexibility does not improve the model's ability to capture subtle variations in the residual patterns induced by different fault types. In fact, the residual space derived from multivariate SVR models appears to be sufficiently linearly separable making LDA more effective.

In conclusion, the classification framework built upon the residuals of the multivariate SVR models, enhanced through a physics-informed preprocessing step, proves to be effective not only in detecting anomalies, but also in discriminating between failure modes. The integration of model-based diagnostics with statistical classification techniques offers a powerful approach to fault identification in wind turbines, enabling early and interpretable failure detection under real-world operating conditions.

6. Final Remarks and Future Work

The main objective of this thesis was to develop and validate anomaly detection methods for wind turbines through vibration monitoring techniques, addressing the challenges posed by confounding factors inherent in their operational conditions. The proposed approach integrates physical modelling, specifically the Betz theory, with both univariate and multivariate regression models based on SVR. Anomaly indices were computed using residuals in the univariate case, and through Mahalanobis distance and One-class SVM scores in the multivariate setting. The final stage of the analysis included a classification task, employing both Linear Discriminant Analysis and Quadratic Discriminant Analysis.
The core idea of the proposed architecture was to model the normal operational behaviour of a wind turbine using SVR, with input features including key physical parameters, such as wind speed, pitch angle, gearbox temperature, rotor speed, and many others, and output features represented by nacelle vibrations, acquired using four accelerometers placed at different points of the nacelle. These vibration signals were used as indicators of potential abnormal behaviour.
Two anomaly detection strategies were evaluated. The first was a univariate method, based on residuals and thresholding techniques informed by training data distributions and ROC curve analysis. The second, multivariate approach relied on the Mahalanobis distance, incorporating the full covariance structure of residuals linked to the healthy data. While the univariate approach yielded satisfactory results, particularly when vibration signals were used as output instead of power produced by the turbine, multivariate analysis demonstrated superior performance in terms of robustness and classification accuracy, especially under varying operating conditions with multiple changing parameters.
The incorporation of the Betz theory proved valuable by introducing physically meaningful features into the SVR model. Specifically, a regression-based approach that models the wind speed-power relationship as a cubic polynomial, referencing the theoretical form suggested by the Betz limit, significantly enhanced anomaly detection in the multivariate case. This contributed to high classification accuracy and clearer separation between normal and faulty conditions.
A further key contribution of this work is the application of discriminant analysis to define class boundaries. The comparison between LDA and QDA showed that QDA outperforms LDA, providing lower FA rates, higher Recall, and reduced class-error. This is mainly due to QDA's ability to model class-specific covariance structures and define non-linear decision boundaries.

The results presented in this thesis confirm that incorporating physical knowledge into machine learning models enhances their diagnostic capabilities. The methodology developed offers a promising framework for vibration-based condition monitoring and fault detection in wind turbines, potentially contributing to more effective predictive maintenance strategies and improved reliability of wind energy systems.

Future Work

This thesis represents one of the few studies applying machine learning techniques to this particular dataset, which contains a wealth of information from three different wind farms. The dataset includes detailed failure conditions for both mechanical and electrical components, as well as various labels that accurately reflect the operational states of the turbines. While the methodology proposed here has proven effective, several avenues remain open for further research and improvement.

First, other regression techniques beyond SVR could be explored. A comprehensive comparison among different models is necessary to fully assess the risk of overfitting and to ensure that the chosen approach generalises well across various operating conditions. Ultimately, the goal of any diagnostic framework in this context is to develop anomaly detection strategies that are capable of generalising the physical and operational behaviour of the machine, so that the same methodology can be applied to other turbines in different locations. This would enable scalable, real-time condition monitoring across entire fleets, particularly critical in industrial and renewable energy sectors such as wind energy.

One of the most important challenges in the field of anomaly detection lies in the ability of algorithms of identify faults independently of operational and environmental variability. These fluctuations can alter the natural dynamic response of a system and may lead to false positives. Therefore, model generalisation is essential: the algorithm must learn the intrinsic characteristics of the system under normal conditions, rather than simply memorising specific data patterns.

A key component of any anomaly detection system is the threshold used to discriminate between normal and anomalous data. The use of static threshold, as implemented in this work, may lead to false alarms during transient or other non-fault related conditions, especially if changes in environmental or operational parameters are not accounted for. As a future improvement, adaptive thresholding techniques could be introduced to account for temporal variations, seasonality, and other non-stationary behaviour in the data.

Although the SVR model offers flexibility and strong regression performance, it does not explicitly model time-dependent behaviours, which are often crucial in fault evolution, particularly in vibration signals. The tuning of the SVR hyperparameters, especially the penalty parameter C plays a critical role in balancing the trade-off between model complexity and error tolerance ε. A more rigorous hyperparameter optimisation, e.g. through grid search combined with k-fold cross-validation on both penalty parameter and error tolerance could enhance model performance and mitigate risks of overfitting or underfitting.

In addition, other machine learning paradigms could be explored. Neural Networks, for instance, provide a fundamentally different modelling approach compared to SVR, offering advantages in learning non-linear and hierarchical feature representations. Autoencoders, in particular, have also shown promise in anomaly detection by compressing input data into a latent space and reconstructing it through a non-linear activation function (e.g. sigmoid function). Deviations between input and reconstruction can then be used as a powerful anomaly indicator.

Given the electromechanical complexity of wind turbines, integrating electrical variables alongside mechanical ones could provide a more holistic view of system health, potentially improving early fault detection performance. This fusion of heterogeneous data sources remains a promising direction for future research.

Ultimately, the key challenge ahead is to reduce MA to ensure uninterrupted energy production and to enhance predictive maintenance strategies. This would enable early fault identification and proactive intervention, thereby minimise costly downtime and extend the operational lifespan of wind turbine systems.

Publication

Paper Conference SURVISHNO 2025

- Marco Gerbino, Alessandro Paolo Daga, Luca Viale, Alessandro Fasana, Luigi Garibaldi. Anomaly Detection in Wind Turbines under Operational Variability via SCADA and Residual Analysis. *Surveillance, Vibrations, Shock and Noise*, Institut Supérieur de Mécanique de Paris [ISAE-SUPMÉCA], May 2025, Paris, France.

References

[1] Stetco, Adrian & Dinmohammadi, Fateme & Zhao, Xingyu & Robu, Valentin & Flynn, D. & Barnes, Mike & Keane, John & Nenadic, Goran. (2018). Machine learning methods for wind turbine condition monitoring: A review. Renewable Energy. 133. 10.1016/j.renene.2018.10.047.

[2] Jon Urmeneta, Juan Izquierdo, Urko Leturiondo. A methodology for performance assessment at system level—Identification of operating regimes and anomaly detection in wind turbines. Renewable Energy, Volume 205, 2023, Pages 281-292, ISSN 0960-1481, https://doi.org/10.1016/j.renene.2023.01.035.

[3] Phong B. Dao, Tomasz Barszcz, Wieslaw J. Staszewski. Anomaly detection of wind turbines based on stationarity analysis of SCADA data. Renewable Energy, Volume 232, 2024, 121076, ISSN 0960-1481. https://doi.org/10.1016/j.renene.2024.121076.

[4] Roelofs, Cyriana & Gück, Christian & Faulstich, S. (2024). Transfer learning applications for autoencoder-based anomaly detection in wind turbines. Energy and AI. 17. 100373. 10.1016/j.egyai.2024.100373.

[5] Tyagi, S., & Panigrahi, S. K. (2017). A Hybrid Genetic Algorithm and Back-Propagation Classifier for Gearbox Fault Diagnosis. *Applied Artificial Intelligence*, *31*(7–8), 593–612. https://doi.org/10.1080/08839514.2017.1413066

[6] Conradi Hoffmann, J. L., Horstmann, L. P., Martínez Lucena, M., Medeiros de Araujo, G., Fröhlich, A. A., & Nishioka, M. H. N. (2021). Anomaly Detection on Wind Turbines Based on a Deep Learning Analysis of Vibration Signals. *Applied Artificial Intelligence*, *35*(12), 893–913. https://doi.org/10.1080/08839514.2021.1966879.

[7] Chesterman, X., Verstraeten, T., Daems, P.-J., Nowe, A., & Helsen, J. (2021). Condition Monitoring of Wind Turbines and Extraction of Healthy Training Data Using an Ensemble of Advanced Statistical Anomaly Detection

Models. In *Proceedings of the Annual Conference of the PHM Society* PHM Society. https://doi.org/10.36001/phmconf.2021.v13i1.2980

[8] Jun Zhan, Chengkun Wu, Xiandong Ma, Canqun Yang, Qiucheng Miao, Shilin Wang. Abnormal vibration detection of wind turbine based on temporal convolution network and multivariate coefficient of variation. Mechanical Systems and Signal Processing, Volume 174, 2022, 109082, ISSN 0888-3270. https://doi.org/10.1016/j.ymssp.2022.109082.

[9] Zhang, Yan & Liu, Wenyi & Wang, Xin & Shaheer, Mirza. (2022). A novel hierarchical hyper-parameter search algorithm based on greedy strategy for wind turbine fault diagnosis. Expert Systems with Applications. 202. 117473. 10.1016/j.eswa.2022.117473.

[10] Marti-Puig, Pere & Blanco-M, Alejandro & Cárdenas, Juan & Cusido, Jordi & Solé-Casals, Jordi. (2019). Feature Selection Algorithms for Wind Turbine Failure Prediction. Energies. 12. 453. 10.3390/en12030453.

[11] Schröder, L.; Dimitrov, N.K.; Verelst, D.R.; Sørensen, J.A. Using Transfer Learning to Build Physics-Informed Machine Learning Models for Improved Wind Farm Monitoring. *Energies* **2022**, *15*, 558. https://doi.org/10.3390/en15020558.

[12] Moreno, S.R., Coelho, L.d.S., Ayala, H.V.H. and Mariani, V.C. (2020), Wind turbines anomaly detection based on power curves and ensemble learning. IET Renew. Power Gener., 14: 4086-4093. https://doi.org/10.1049/iet-rpg.2020.0224.

[13] Pandit, R.; Kolios, A. SCADA Data-Based Support Vector Machine Wind Turbine Power Curve Uncertainty Estimation and Its Comparative Studies. *Appl. Sci.* **2020**, *10*, 8685. https://doi.org/10.3390/app10238685

[14] Zhang S, Robinson E, Basu M. Wind turbine condition monitoring based on three fitted performance curves. *Wind Energy*. 2024; 27(5): 429-446. doi:10.1002/we.2859

[15] Vidal, Y.; Pozo, F.; Tutivén, C. Wind Turbine Multi-Fault Detection and Classification Based on SCADA Data. *Energies* **2018**, *11*, 3018. https://doi.org/10.3390/en11113018

[16] Abdallah, Imad & Dertimanis, Vasilis & Mylonas, Charilaos & Tatsis, Konstantinos & Chatzi, Eleni & Dervilis, Nikolaos & Worden, Keith & Maguire, A. (2018). Fault diagnosis of wind turbine structures using decision tree learning algorithms with big data. 10.1201/9781351174664-382.

[17] Bangalore, Pramod and Lina Bertling Tjernberg. "An Artificial Neural Network Approach for Early Fault Detection of Gearbox Bearings." *IEEE Transactions on Smart Grid* 6 (2015): 980-987.

[18] Pandit R, Astolfi D, Hong J, Infield D, Santos M. SCADA data for wind turbine data-driven condition/performance monitoring: A review on state-of-art,

challenges and future trends. *Wind Engineering*. 2022;47(2):422-441. doi:10.1177/0309524X221124031

[19] Viale, L.; Daga, A.P.; Fasana, A.; Garibaldi, L. From Novelty Detection to a Genetic Algorithm Optimized Classification for the Diagnosis of a SCADA-Equipped Complex Machine. *Machines* **2022**, *10*, 270. https://doi.org/10.3390/machines10040270

[20] Chandola, Varun & Banerjee, Arindam & Kumar, Vipin. (2009). Anomaly Detection: A Survey. ACM Comput. Surv. 41. 10.1145/1541880.1541882.

[21] The Elements of Statistical Learning. T. Hastie, R. Tibshirani, and J. Friedman. *Springer Series in Statistics Springer New York Inc., New York, NY, USA,* (*2001*).

[22] Deisenroth, M. P., Faisal, A. A., & Ong, C. S. (2020). *Mathematics for Machine Learning*. Cambridge: Cambridge University Press.

[23] Tchakoua, P.; Wamkeue, R.; Ouhrouche, M.; Slaoui-Hasnaoui, F.; Tameghe, T.A.; Ekemb, G. Wind Turbine Condition Monitoring: State-of-the-Art Review, New Trends, and Future Challenges. *Energies* **2014**, *7*, 2595-2630. https://doi.org/10.3390/en7042595.

[24] Daga, A. P., Fasana, A., Garibaldi, L., & Marchesiello, S. (2020). On the use of PCA for Diagnostics via Novelty Detection: interpretation, practical application notes and recommendation for use. *PHM Society European Conference*, *5*(1), 13. https://doi.org/10.36001/phme.2020.v5i1.1241.

[25] Natili, F.; Daga, A.P.; Castellani, F.; Garibaldi, L. Multi-Scale Wind Turbine Bearings Supervision Techniques Using Industrial SCADA and Vibration Data. *Appl. Sci.* **2021**, *11*, 6785. https://doi.org/10.3390/app11156785.

[26] Castellani, F.; Garibaldi, L.; Daga, A.P.; Astolfi, D.; Natili, F. Diagnosis of Faulty Wind Turbine Bearings Using Tower Vibration Measurements. *Energies* **2020**, *13*, 1474. https://doi.org/10.3390/en13061474.

[27] Marco A.F. Pimentel, David A. Clifton, Lei Clifton, Lionel Tarassenko. A review of novelty detection. Signal Processing, Volume 99, 2014, Pages 215-249, ISSN 0165-1684. https://doi.org/10.1016/j.sigpro.2013.12.026.

[28] Viale, L.; Daga, A.P.; Fasana, A.; Garibaldi, L. From Novelty Detection to a Genetic Algorithm Optimized Classification for the Diagnosis of a SCADA-Equipped Complex Machine. *Machines* **2022**, *10*, 270. https://doi.org/10.3390/machines10040270.

[29] Vibration Monitoring: Gearbox identification and faults detection / Daga, ALESSANDRO PAOLO. - (2019 Oct 17), pp. 1-262.

[30] Fasana, A., & Marchesiello, S. (2006). Meccanica delle vibrazioni. Ed. CLUT.

[31] Dhiman, H.S.; Deb, D.; Carroll, J.; Muresan, V.; Unguresan, M.-L. Wind Turbine Gearbox Condition Monitoring Based on Class of Support Vector Regression

Models and Residual Analysis. *Sensors* **2020**, *20*, 6742. https://doi.org/10.3390/s20236742.

[32] Gück, C.; Roelofs, C.M.A.; Faulstich, S. CARE to Compare: A Real-World Benchmark Dataset for Early Fault Detection in Wind Turbine Data. *Data* **2024**, *9*, 138. https://doi.org/10.3390/data9120138.

[33] Randall, R.B. (2010). Vibration-based Condition Monitoring: Industrial, Aerospace and Automotive Applications. Vibration-based Condition Monitoring: Industrial, Aerospace and Automotive Applications. 10.1002/9780470977668.

[34] Thomas Hofmann. Bernhard Schölkopf. Alexander J. Smola. "Kernel methods in machine learning." Ann. Statist. 36 (3) 1171 - 1220, June 2008. https://doi.org/10.1214/009053607000000677.

[35] Gück, C., & Roelofs, C. (2024). Wind Turbine SCADA Data for Early Fault Detection (v1.0) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.10958775.

[36] Fazylova, A.; Tultayev, B.; Iliev, T.; Stoyanov, I.; Kabasheva, M.; Kosunalp, S. Experimental Study of an Industrial Data Transmission Network in the Automatic Control System of a Wind Turbine. *Machines* **2024**, *12*, 746. https://doi.org/10.3390/machines12110746.

[37] Lundquist, Julie K., Clifton, Andrew J., Dana, Scott, Huskey, Arlinda, Moriarty, Patrick J., Van Dam, Jeroen J., & Herges, Tommy (2019). Wind Energy Instrumentation Atlas. https://doi.org/10.2172/1513195.

[38] Keller, Jonathan & Wallen, Robb. (2016). Gearbox Reliability Collaborative Phase 3 Gearbox 3 Test Plan.

[39] Upadhyay, Dhaval & Sampalli, Srinivas. (2019). SCADA (Supervisory Control and Data Acquisition) Systems: Vulnerability Assessment and Security Recommendations. Computers & Security. 89. 101666. 10.1016/j.cose.2019.101666.

[40] García Márquez, Fausto Pedro & Tobias, Andrew & Pinar Pérez, Jesús María & Papaelias, Mayorkinos. (2012). Condition Monitoring of Wind Turbines: Techniques and Methods. Renewable Energy. 46. 169–178. 10.1016/j.renene.2012.03.003.

[41] Jenkins, N., Burton, A., Sharpe, D., & Bossanyi, E. (2001). *Wind Energy Handbook*. John Wiley & Sons Ltd.

[42] Manwell, J.F., McGowan, J.G. and Rogers, A.L. (2009) Wind Energy Explained: Theory, Design and Application. Wiley, Chichester. https://doi.org/10.1002/9781119994367

[43] Schaarup, J. (Ed.) (2002). *Guidelines for design of wind turbines. 2. ed.* Risø National Laboratory.

[44] Puntanen, Simo. (2013). Handbook of Regression Analysis by Samprit Chatterjee, Jeffrey S. Simonoff. International Statistical Review. 81. 10.1111/insr.12020_22.

[45] 1. Cristianini N, Shawe-Taylor J. *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge University Press; 2000.

[46] Larry M. Manevitz and Malik Yousef. 2002. One-class svms for document classification. J. Mach. Learn. Res. 2 (3/1/2002), 139–154.

[47] Ghorbani, Hamid. (2019). MAHALANOBIS DISTANCE AND ITS APPLICATION FOR DETECTING MULTIVARIATE OUTLIERS. Facta Universitatis Series Mathematics and Informatics. 34. 583. 10.22190/FUMI1903583G.

[48] Sørbø, S., Ruocco, M. Navigating the metric maze: a taxonomy of evaluation metrics for anomaly detection in time series. *Data Min Knowl Disc* **38**, 1027–1068 (2024). https://doi.org/10.1007/s10618-023-00988-8.

[49] Chicco, D., Warrens, M.J. and Jurman, G. (2021) The Coefficient of Determination R-Squared Is More Informative than SMAPE, MAE, MAPE, MSE and RMSE in Regression Analysis Evaluation. PeerJ Computer Science, 7, e623. https://doi.org/10.7717/peerj-cs.623

[50] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, Apr. 2016.

[51] Hodson, T. O.: Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not, Geosci. Model Dev., 15, 5481–5487, https://doi.org/10.5194/gmd-15-5481-2022, 2022.

[52] Chai, T. and Draxler, R. R.: Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature, Geosci. Model Dev., 7, 1247–1250, https://doi.org/10.5194/gmd-7-1247-2014, 2014.