The Impact of Feature Causality on Normal Behaviour Models for SCADA-based Wind Turbine Fault Detection

Telmo Felgueira
telmo.felgueira@ist.utl.pt

Instituto Superior Técnico, Lisboa, Portugal

June 2019

Abstract

The cost of wind energy can be reduced by using SCADA data to detect faults in wind turbine components. Normal behavior models are one of the main fault detection approaches, but there is a lack of consensus in how different input features affect the results. Also, most works use ANNs, which are known to be black-box models with high computational costs. In this work, a new taxonomy based on the causal relations between the input features and the target is presented. Based on this taxonomy, the impact of different input feature configurations on the modelling and fault detection performance is evaluated. To this end, a framework that formulates the detection of faults as a classification problem is also presented. Finally, GBMs will be tested as an alternative to ANNs.

Keywords: causal, NBM, turbine, fault, SCADA

1. Introduction

In 2018, global energy-related CO_2 emissions reached a historic high of 33.1 gigatonnes. These emissions are caused by the burning of fossil fuels, mainly natural gas, coal and oil, which accounted for 64% of global electricity production in this same year [1]. Greenhouse gases like CO_2 are responsible for climate change which threatens to change the way we have come to know Earth and human life. For the previous reasons, there has been a global effort to shift from a fossil fuel based energy system towards a renewable energy one. In fact, it is expected that by 2050 wind energy will represent 14% of the world's total primary energy supply [2].

The operation and maintenance costs of Wind Turbines (WTs) can account for up to 30% of the cost of wind energy [3]. This happens because while generators in fossil fuel power plants operate in a constant, narrow range of speeds, WTs are designed to operate under a wide range of wind speeds and weather conditions. This means that stresses on components are significantly higher, which increases the number of failures and consequently the maintenance costs [4].

There have been recent efforts to monitor and detect incipient faults in WTs by harvesting the high amounts of data already generated by their Supervisory control and data acquisition (SCADA) systems, which, in turn, enables the wind farm owners to employ a predictive maintenance strategy. In fact, it is expected that by 2025 new predictive maintenance strategies can reduce the cost of wind energy by as much as 25% [5]. One of the main methods for monitoring the condition of WTs is building Normal Behaviour Models (NBMs) of the component temperatures. The fundamental assumption behind the use of NBMs is that a fault condition is normally characterized by a loss of efficiency, which results in increased temperatures. By using SCADA data to build a model of the temperatures of the WT components, one can calculate the residuals, which are the difference between the real values measured by the sensors and the predicted values by the model. These residuals can be used to detect abnormally high temperatures that may be indicative of an incipient fault.

Firstly, it's important to clarify that NBMs can be evaluated in terms of how well they model the target temperature and in terms of how well they detect faults. The first corresponds to evaluating how low are the residuals during healthy periods of time, while the second corresponds to evaluating if there is an increase in the residuals before a failure. Regarding the literature, multiple works [6–8] have reported good results using NBMs, being able to predict failures in WT components months in advance. In these works, the authors used as features active power, nacelle temperature and lagged values of the target temperature, thus including autoregressive properties into the model. In [9] and [10] the authors noted that although the use of autoregressive features resulted in better temperature modelling performance it also resulted in worse fault detection performance. But there have been works that obtained the opposite result, where including autoregressive features improved both the temperature modelling performance of the model and the fault detection performance, such as [11, 12]. Another important result was obtained in [13] and [14], which indicated that using features that are highly correlated with the target also increased the modelling performance but decreased the fault detection performance of the model. Nonetheless, these type of features are still used in a variety of works today, such as [10, 15–17].

Summarizing, there is a lack of consensus regarding which input features should be used, existing significant variation between works. The main reason behind this is the lack of consistent case studies that evaluate the impact of different features on both the temperature modelling and fault detection performances. It should also be noted that in NBMs it's not trivial that the more features the model has the better its fault detection performance will be. This happens because the model is being trained to minimize the temperature modelling error and not the fault detection one. Having this in mind, this work will present a new feature taxonomy to distinguish different input feature types. Then, the impact of these input feature types on the temperature modelling and fault detection performances will be evaluated.

It's also important to note that all mentioned works have used Artificial Neural Networks (ANNs) or ANN-based algorithms to build the NBMs. This may be due to the existing domain knowledge of the non-linear relationships between the input features and the target. Since ANNs are known for their highly non-linear modelling capabilities, they are good candidates. Nonetheless, these works have also criticized their high computational costs, time-consuming optimization and low interpretability. The latter is specially important in this type of industrial application, where there is skepticism towards black-box models. This raises the need for other solutions. A good candidate are Gradient Boosting Machines (GBMs), known for their highly non-linear modelling capabilities, while having considerably lower computational costs and being more robust to hyperparameter optimization [18]. GBMs also have higher interpretability due to being tree-based models. Also, GBMs have obtained excellent results in time series modelling, as shown in works [19] and [20]. For these reasons, this work will assess the use of GBMs for building NBMs.

Finally, evaluating the fault detection performance of different models is not as trivial as evaluating their temperature modelling performance. In fact, there is no standard in the literature regarding how to evaluate fault detection performance. This happens because of the inherent nature of the fault detection problem, in which there is rarely groundtruth. Indeed, there is data of when the failure happened, but there is no information regarding when the fault state started, making it not trivial to formulate as a classification problem. Hence why the majority of the literature evaluates the fault detection results by visual inspection, observing the increase in the residuals before the failure. This is problematic, because comparisons between different models will be highly subjective. Having this in mind, this work will also present a formulation of the detection of faults as a classification problem.

2. Methods

The data used in this work comes from a wind farm composed of 16 turbines, from the beginning of 2008 to the end of 2013. During the years of 2012 and 2013 there were a total of nine failures related with the gearbox bearing of the WTs. Hence, this will be the component for which a NBM will be trained. It should also be noted that all the work was developed in Python 3, using Pandas [21] for data processing and Plotly [22] for data visualization.

2.1. Gradient Boosting Machines and Training

The NBMs in this work will be based in GBMs, which are a machine learning technique that uses a prediction model in the form of an ensemble. This means that it combines multiple simple models into a single composite model. In boosting terminology, the simple models are called weak learners. In this work, as is standard for most problems, the weak learners will be decision trees.

Given the input features vector x, the ensemble $F_M(x)$ will be composed of M weak learners of the form $f_m(\mathrm{x})$, which will have the predictions $\hat{y}$ for the target y. This is formalized in equation 1. To understand the process of building the ensemble we can imagine that the starting weak learner simply predicts the mean of the observations, such that $f_0(x) = \overline{y}$. That means the residuals for that iteration will be given by Equation 2. Now, another weak learner, $f_1(x)$, will be added to the composite model, and ideally it would make $F_1(x)$ able to predict y as Equation 3 shows. For this to happen, the new weak learner must be equal to the residuals, as demonstrated in Equation 4. Note that in practice the model would need more weak learners, and it would never equal the target values. Nonetheless, the main idea is that in GBMs the added models are trained on the residuals of the previous model. For the general case it can be summarized as in Equation 5, in which it's also been added the learning rate η, which is a hyperparameter used to prevent overfitting, so that each added weak learner has less of an effect on the composite model.

$$F_M(x) = \sum_{m=0}^{M} f_m(x) = \hat{y} \tag{1}$$

$$r_0 = y - f_0(x) = y - \overline{y} \tag{2}$$

$$F_1(x) = f_0(x) + f_1(x) = y \tag{3}$$

$$f_1(x) = y - f_0(x) = r_0 \tag{4}$$

$$\hat{y}_m = \hat{y}_{m-1} + \eta r_{m-1} \tag{5}$$

Regarding the dataset division, the models were trained with data from the beginning of 2008 to the end of 2011 and tested on data from 2012 and 2013. Periods with faults will be removed from the training data so the model does not learn abnormal behaviour. In terms of implementation, LightGBM [23] will be used due to its high computational performance. In terms of optimization, the year of 2011 will be used as a validation set when choosing the number of trees for each model by early stopping. Note that no exhaustive hyperparameter optimization was performed, so all models will use the same hyperparameters besides the number of trees.

2.2. Feature Taxonomy

As mentioned in the Introduction, there have been works which indicated that using features that are highly correlated with the target increases temperature modelling performance but may decrease the fault detection performance of the model [13, 14]. For example, if the gearbox bearing temperature is being modelled, it's expected that when it gets hotter the gearbox oil temperature will also get hotter, due to heat transfer. Indeed, it is intuitive that the gearbox oil temperature is highly correlated with the gearbox bearing temperature and thus will be important for modelling. But the problem is that a state of fault is characterized by overheating in the gearbox bearing temperature, and again, due to heat transfer, the gearbox oil temperature will also be hotter than expected. This means that if the gearbox oil temperature is being used to model the gearbox bearing temperature, the model may model abnormal behavior and thus not be able to detect the incipient fault. Having this in mind, the present work hypothesizes that the decrease in fault detection performance is not due to the fact that the features are highly correlated with the target temperature, which is subjective since there is significant correlation between rotor speed and the gearbox bearing temperature. In fact, we suggest that what decreases the fault detection performance is using features that have an interdependence with the target, so not only is the target dependent on them, they are also dependent on the target. This happens for all temperatures in the drivetrain, since there is heat transfer between all of them.

Although the ideas behind the previous hypothesis are intuitive, it's important to make it more objective by using a clearer nomenclature. For this reason, we will present a new taxonomy based on Econometric Causality [24] and Causal Inference [25], which distinguishes features based on their causal relations with the target. Basically, if the target is causally dependent of the features, they are considered causal features. On the other hand, if there is a causal interdependence between the feature and the target, as in they are dependent on each other, they are considered simultaneity features.

As was shown in the previous example, these causal relations can be assumed since we have domain knowledge of the physical system. The present work suggests the causal diagram presented in Figure 1, where the arrows represent causal relations. If the arrow is double-pointed, it means that there is interdependence between the variables. It should also be clarified that mediators are also causal features. The only difference is that they are not the original cause, in fact they mediate the causal effect from other causal features. For example, the gearbox bearing temperature depends on the rotor speed, but the rotor speed depends on the wind speed. This means that the origin of the causal effect is the wind speed, thus meaning that this is a causal feature, while the rotor speed is a mediator of this causal relation. For the purpose of the work, mediators will be considered causal features, but there is interest in further exploring how this intricacies can affect the NBM both in terms of temperature modelling and fault detection performances.

The introduced taxonomy clarifies some details from previous works. For example, usually the nacelle temperature was not considered a feature highly correlated with the target, and the works that criticized the use of highly correlated features actually used the nacelle temperature [13]. According to this taxonomy, both the nacelle temperature and the gearbox oil temperature are simultaneity features, which means that if the gearbox oil temperature does negatively affect fault detection performance, then it's expected that nacelle temperature also does.

Regarding the use of autoregressive features, these are clearly causal features, since there is a temporal relation that prevents the future values from affecting past values. Nonetheless, as noted in the literature, the impact of these features in model performance is not consensual. For example, if there is a developing fault and the gearbox

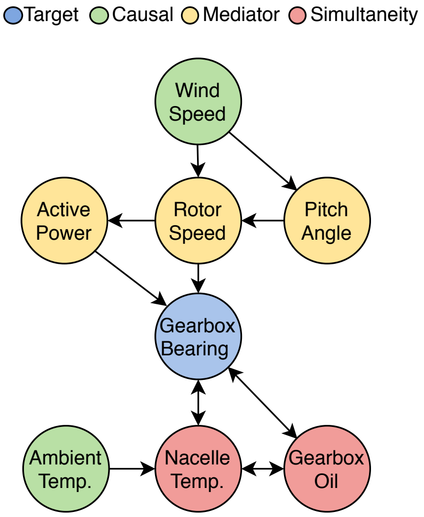


Figure 1: Causal diagram for the gearbox bearing temperature.

bearing is overheating, then the use of previous values that were already hotter than they should be to model the present temperature may result in the NBM modelling abnormal behavior. In fact, using autoregressive features in an NBM can also be seen as what is being modelled is not the actual gearbox bearing temperature, but the temperature rate of change. Since the model has knowledge of the previous timestamps, it is expected to learn the normal rate of change of temperature. Thus, the model should be mostly insensitive to changes in the mean value of the target temperature. This is problematic if the fault is characterized by a change in the mean temperature for the given conditions, but if the fault is also characterized by an abnormal rate of change in the temperature then the autoregressive model should be able to detect it. This indicates that the impact of autoregressive features may depend on the type of fault, which may also explain the non-consensual results obtained in the literature.

2.3. Feature Configurations

Based on the previous taxonomy, different models will be defined based on their input feature configuration. The simplest model that will be tested is the Causal Normal Behaviour Model (CNBM), which only uses causal features. These are determined based on domain knowledge and will be: rotor speed, active power, pitch angle, wind speed and ambient temperature. All these features characterize the operation regimes of the WT. Another input feature configuration, the Simultaneous Normal Behaviour Model (SNBM), will consist of using both causal features and simultaneity features. Regarding the simultaneity features, since the ones most used in the literature are nacelle temperature and gearbox oil temperature, these will be the ones tested in this work on models SNBM1 and SNBM2 respectively. Another model will be the autoregressive version of the CNBM, the Autoregressive Causal Normal Behaviour Model (ACNBM). Then, the autoregressive versions of the SNBMs, the Autoregressive Simultaneous Normal Behaviour Model (ASNBM), will be tested in both variants: ASNBM1 and ASNBM2. A summary of the input features used in each model is presented in Table 1

Table 1: The defined models and the corresponding input features.

Model	Causal Features	Nacelle Temp.	Gearbox Oil Temp.	AR
CNBM	X			
SNBM1	X	X		
SNBM2	X		X	
ACNBM	X			X
ASNBM1	X	X		X
ASNBM2	X		X	X

2.4. Fault Evaluation Framework

Before the actual evaluation framework one must obtain alarms from the residuals. Indeed, this can be done by simply using thresholds, in which all the values above the threshold are alarms. But in the context of predictive maintenance it's not necessary to know if there is an alarm at every sample of the signal. In fact, for this work knowing if there is an alarm each day is sufficient. For this reason, the alarms that will be used in the framework will be generated from a daily mean resample of the residuals. These will be compared with the labels, by making use of the evaluation framework that will be described.

To develop an evaluation framework for fault detection, one must first formulate it as a binary classification problem where there are two labels: fault and no-fault. Since there is no information regarding the fault state of the component, only the date of failure, it was defined with the wind farm owners that for the failures studied in this work it can be assumed that a fault state would be present at most 90 days before the failure. It was also defined that for the alarms to be useful they should be triggered at least 15 days before the failure. This means that to be considered a True Positive (TP) the alarm must be triggered between 90 and 15 days before the failure. Figure 2 presents a schematic example of the previously described problem formulation. Taking this example, it is important to note that the number of alarms triggered in the prediction window is not relevant, they are all aggregated as 1 TP. The main reason for this, is that if the aggregation is not done, then 4 alarms for the same failure would count as much as 4 detected failures with 1 alarm each. This clearly is not what is intended of the framework, since 1 alarm should be enough to motivate an inspection, and detecting 4 failures with 1 alarm outweighs detecting 1 failure with 4 alarms. Finally, it is also important to note that alarms triggered less than 15 days before the failure are not considered False Positives (FPs), since there is indeed a fault state, it simply is not relevant, so they are considered True Negatives (TNs).

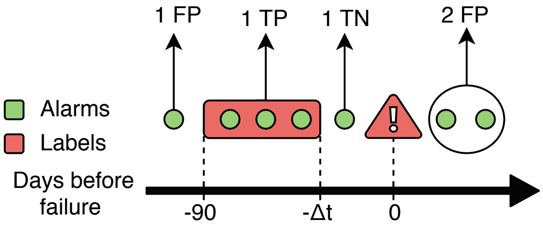


Figure 2: Schematic example of the fault detection classification problem formulation.

It should be noted that for the implementation of the framework with the total 16 turbines, the thresholds values ranged from -2 to 4, with incremental steps of 0.1. The minimum prediction windows were 1, 10, 20, 30, 40, 50, 60, 70 and 80 days. Also, this process will be done for each of the six models. This results in a total of 51840 combinations, which can be considerably time-consuming to compute. Having this in mind, the evaluation framework was implemented making use of Dask [26], which provides parallel and distributed functionalities to Pandas. To take full advantage of Dask's distributed capabilities, the results were calculated using a cluster computing framework in the cloud, with a total of 30 workers with eight cores each, thus allowing 240 tasks to be performed in parallel and obtaining considerably better computational time performance.

2.5. Classifier Evaluation

Although the confusion matrix provides all the information related with the fault detection performance of the model, the objective is to summarize this into a single classification metric. Here it's important to remember that although the majority of the literature only does visual residual analysis, there have been some works that have evaluated the results as a classification problem. For example, in [15] the author used the Receiver operating characteristic (ROC) curve to evaluate the performance of each model. But it's also important to remind that this problem is clearly an unbalanced one, since the number of no-faults labels is notably higher than the number of fault labels. From this results that the use of the ROC curve and other more common metrics such as accuracy are not adequate, since they are not robust to unbalanced data. On the other hand, metrics such as precision and recall are significantly more robust to unbalanced data [27]. The definition of precision and recall is presented in Equations 6 and 7, and it should be noted that they have an interpretable meaning. Precision corresponds to how many of the triggered alarms were true, and recall corresponds to how many of the failures were detected. For each threshold corresponds a precision and a recall, so for varying thresholds one can construct a precision and recall curve. Each precision and recall curve can be summarized into a metric, the Area Under the Curve (AUC), which corresponds to the area under the curve. Given that the precision and recall curve is a discrete function, the AUC can be calculated by Equation 8, where k corresponds to the different thresholds of the framework. This will be the metric used to evaluate fault detection performance.

$$P = \frac{tp}{tp + fp} \tag{6}$$

$$R = \frac{tp}{tp + fn} \tag{7}$$

$$AUC = \sum_{k=1}^{N} P(k) \Delta R(k) \tag{8}$$

3. Temperature Modelling Results

In Figure 3 is presented a period of time where an healthy turbine worked under different regimes, such as high and low power production, as can be seen by the active power signal, and also during braking, as can be seen by the pitch angle signal. Regarding the predictions of the models, in this case study the CNBM and the ACNBM are being evaluated, and as can be seen both models are able to follow the true signal during the majority of time, even for different turbine loads. But it's important to note that the predictions of the CNBM are significantly worse when the blades pitch to around 95°to stop the turbine. For example, during the morning of July 31st there are two periods of time where the WT stopped by using the pitch angle brake system, and the CNBM predicts that the temperature of the gearbox bearing would be lower than what it really is. This is problematic, since it would lead to false positives during the fault detection, because the real temperature is below the predicted one, not due to the existence of a fault but due to the model not learning the braking regime.The reason behind this may be due to the fact that the WT being stopped by the pitch brake system is not a common event, which means it is under represented in the dataset. This hypothesis is supported by the fact that in the training dataset only 1% of data corresponds to when the pitch angle is above 90°. This means that

Table 2: Regression evaluation metrics for the different models on the train and test sets.

Model	Train Results (°C)			Test Results (°C)		
	MAE	RMSE	SD	MAE	RMSE	SD
CNBM	1.05	1.63	1.25	1.36	1.75	1.11
SNBM1	1.03	1.61	1.23	1.31	1.74	1.15
SNBM2	0.39	0.56	0.40	0.52	0.73	0.51
ACNBM	0.55	0.85	0.65	0.63	0.94	0.69
ASNBM1	0.51	0.80	0.62	0.60	0.91	0.68
ASNBM2	**0.33**	**0.48**	**0.35**	**0.41**	**0.59**	**0.42**

Table 3: Algorithm and features used in the NBMs and corresponding results for works in the literature.

Work	Input Features				Test Results (°C)		
	Causal	Nacelle Temp.	Gearbox Oil Temp.	AR	MAE	RMSE	SD
[28]	X				2.15	2.93	2.88
[29]	X	-	-	-	0.663	-	-
[15]	X	X	X		-	1.22	-
[30]	X	X	X		-	-	**1.3**
[6]	X	X		X	-	1.23	-
[13]	X	X	X	X	**0.44**	0.77	-
[10]	X	X	X	X	-	**0.31**	-

the models wouldn't be able to learn this behavior as well as the most represented ones. On the other hand, the ACNBM is able to predict the temperature with much less error in this regime. This may be due to the fact that the ACNBM has autoregressive features, which considerably simplify the prediction task, thus explaining the notably better results during this regime, independently of it being under-represented. Another possible explanation is that the braking of the turbine is exactly the time when it is harder to predict the temperature behavior from causal features, since it results in the stopping of the rotating components, making it the regime in which these causal features provide the least predictive power to the model. From this, results that in this regime the target temperature mostly depends on its previous values, hence why the autoregressive models performs better.

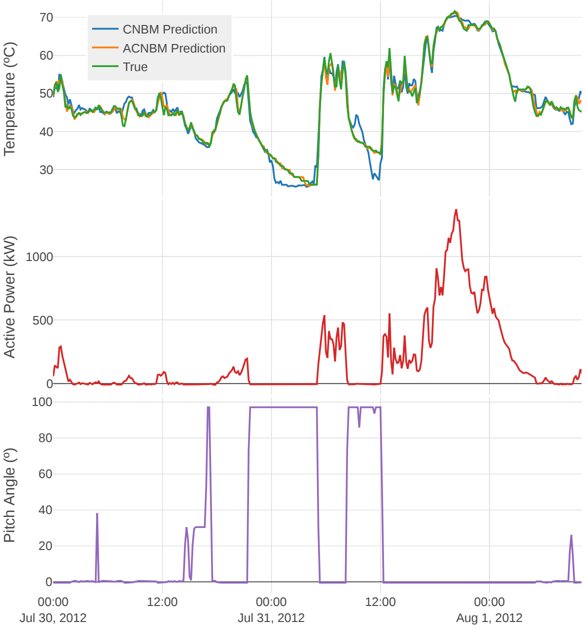
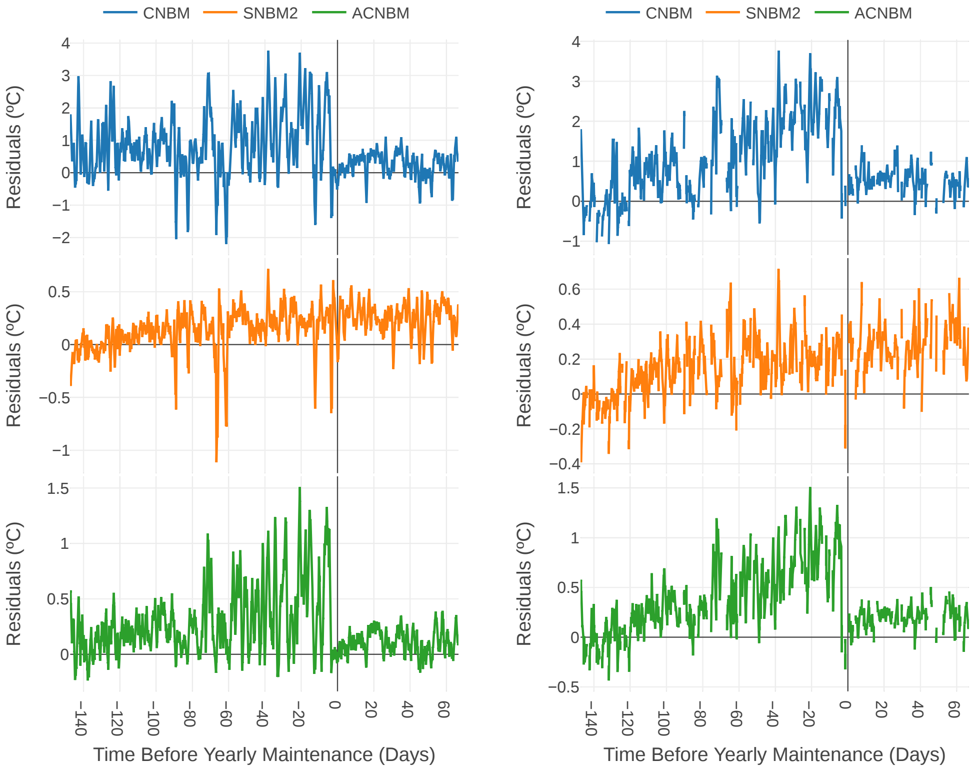


Figure 3: Target temperature predictions for the CNBM and ACNBM against the true values.

The results for all the healthy periods of the total WTs are presented in Table 2 for the train and test sets, in terms of Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and Standard Deviation (SD). As expected, the ASNBM2 obtains the best performance on all for metrics for both the training and test sets, since it is the model with more features. The results also indicate that the gearbox oil temperature provides more predictive power to the model than the autoregressive features, hence why the SNBM2 has a better performance than the ACNBM. Also, the nacelle temperature doesn't seem to provide significant predictive power to the model, since the CNBM obtains similar results to the SNBM1. This may be due to the fact that all the relevant information that the nacelle temperature contains is already explained by the ambient temperature. Finally, these results also confirm the point previously raised, that autoregressive features and highly target correlated simultaneity features such as the gearbox cooling oil temperature increase the temperature modelling



(a) Without filter

(b) With active power filter

Figure 4: Residuals during a fault state for the CNBM, SNBM2 and ACNBM.

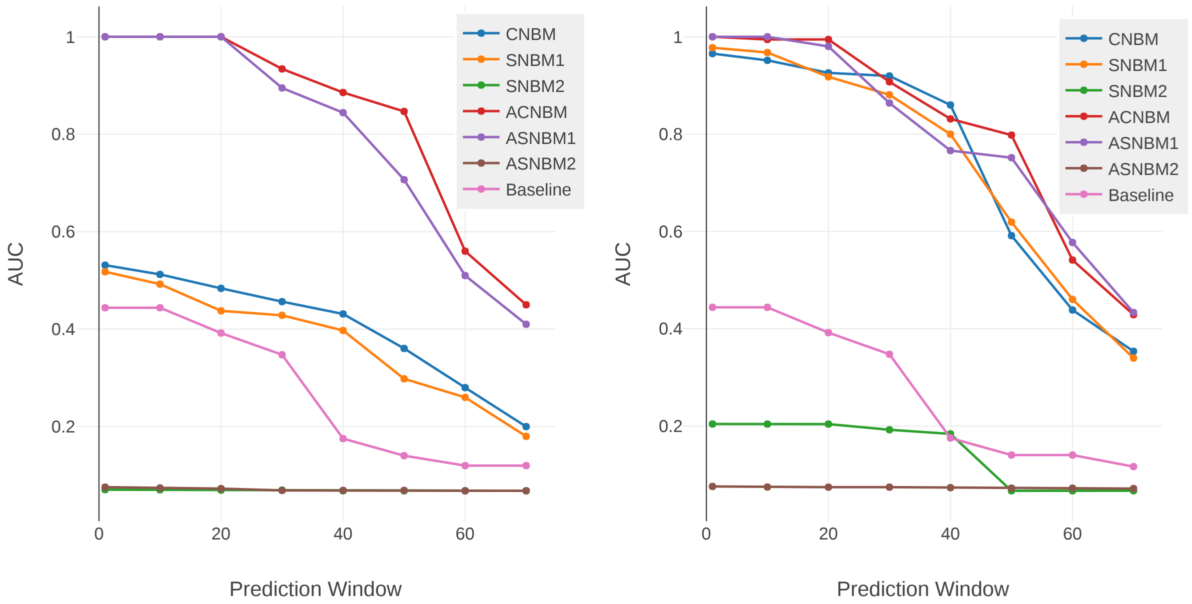
performance of the model. This is due to them modelling better specific regimes, as was seen in Figure 3, but also due to modelling better the overall behavior of the component.

It's also important to compare these results with the ones reported in the literature. To this end, in Table 3 are presented the results from works in the literature, with the corresponding input features of the NBMs. Due to the inexistence of a standard dataset in this area, comparisons between works are always susceptible to some degree of subjectivity. But since the datasets are all from the SCADA system, the major part of the difference in the results should originate from the input features and the algorithm used. Having this in mind, it's interesting to note that [28] is the only model to only use causal features and the results obtained are significantly worse than the ones obtained by the CNBM in this work, which also only uses causal features. Regarding SNBMs, the results from [15] and [30] are also worse than the ones obtained in this work with the SNBM2. Finally, in terms of autoregressive models, the ASNBM2 obtained better results than [13] but worse results than [10]. In general, these results indicate that GBMs can obtain similar results to ANNs, thus being a potential alternative for significantly lower computational costs. Furthermore, tree-based methods such as GBMs have increased model interpretability, which is important for industry related applications, such as predictive maintenance, since there is skepticism regarding black-box models.

4. Fault Detection Results

The results presented in Figure 4a show the residuals of each model during a state of fault. The models CNBM and ACNBM clearly show an increase in the residuals previously to the failure and a decrease after the corresponding maintenance. Indeed, these models detect a fault state previously to the failure. On the other hand, the SNBM2 doesn't detect the fault state, showing no increase nor decrease of the residuals. This indicates that this fault resulted in an overheating of the gearbox bearing and in consequence the gearbox oil temperature also increased. Hence why using the gearbox oil temperature as an input feature to the model resulted in a leak of information regarding the fault state of the component, thus making the model predict abnormal behavior and not detect the fault.

Regarding the CNBM and the ACNBM, both show an increase in the residuals at least 70 days before the failure. What is interesting is that the residuals seem to indicate that the CNBM could have actually predicted the failure even earlier, having spikes up to 140 days before the failure. This



(a) Without filter

(b) With active power filter

Figure 5: Evolution of the AUC over the prediction windows for the different models.

motivated a more careful inspection of the residuals which resulted that, in fact, these residual spikes are related with the regimes that the CNBM did not learn, as previously noted in Figure 3. This was expected, since the maximum predictive window should be 90 days, according to the established groundtruth. This also confirms the previous assumption that the regimes not learned by the CNBM may lead to false positives.

Regarding the regimes not learned by the CNBM, it should be reminded that they happen mostly when the turbine is shutting down. This makes them the least relevant for fault detection since it's when the rotating components are stopped, and thus the temperature variations are not necessarily related with efficiency changes, which are the basis of using NBMs for fault detection. Thus, a simple approach to improve the results, as was already proposed in [15], is to apply an active power filter. The majority of these regimes happened when the WT had active power below 200kW, so this was the value used, below which the residuals are filtered out. In Figure 4b are presented the results with the active power filter. It's interesting to note that the residual spikes of the CNBM at 140 days and 120 days before the failure disappeared, indicating they were indeed false positives related with the turbine shutting down.

4.1. Evaluation Framework Results

The visual residual analysis provided intuition in how well each model predicts failures. But as motivated in the Introduction, this analysis is both subjective and time-consuming. Having this in mind, the results of the developed automated evaluation framework will now be analysed. Before that, it should be noted that a baseline was defined that consists of setting different thresholds on the distribution of the target temperature and obtaining the corresponding precision and recall.

In Figure 5 is presented the evolution of the AUC for the different models over the various prediction windows, with and without the active power filter. Regarding 5a, the results indicate that the ACNBM obtains the best performance for all predictive windows. Indeed, its non-autoregressive counterpart, the CNBM, obtains considerably worse results. This may be due to the FPs that the CNBM has, as was seen in the visual residual analysis. This leads to a lower precision and consequently a lower AUC. It's also interesting to note that the models with the nacelle temperature obtain worse results, indicating that although this feature doesn't impact temperature modelling performance it does decrease fault detection performance. Finally, the models with the gearbox oil temperature clearly obtain the worse results. This is aligned with the results from the visual residual analysis, where the SNBM2 did not detect the fault state. The results with the active power filter are presented in Figure 5b. These results confirm the results from the visual inspection. The CNBM clearly improves the performance with the filter, being close to the autoregressive models. This also confirms the assumptions that the low AUC was due to the model not capturing certain regimes.

5. Conclusions

The majority of the literature uses ANNs for building NBMs. In this work, it was shown that GBMs obtain competitive temperature modelling results. This is relevant because GBMs are known to have lower computational costs and also higher model interpretability. It was also developed a taxonomy to categorize input features into different types based on their causal relation with the target temperature. This allowed to evaluate how different input feature affect the performance of the model. Regarding temperature modelling performance, causal models are able to follow the target temperature during the majority of the time, but have significant error during certain regimes. The addition of the autoregressive features makes the previous model able to capture those regimes, leading to a considerably better temperature modelling performance. Furthermore, the addition of the nacelle temperature doesn't have a significant impact in the temperature modelling performance of the model, while the gearbox oil temperature notably increases it, even more than the autoregressive features.

Then, detection of faults was formulated as a classification problem and an evaluation framework was developed using classification metrics for unbalanced datasets. The results from this framework confirmed those obtained by visual inspection, indicating that this is a good alternative which is more objective and involves less manual work. Regarding the fault detection performance of the models, the results indicated that the causal model obtains better AUC than the baseline, but it has significant false positives due to the regimes it did not learn as well. The addition of autoregressive features reduces these false positives thus increasing fault detection performance. Indeed, the model with the autoregressive features was able to predict all failures without false positives up to 20 days before the failure. As previously motivated, the effect of using autoregressive features may depend on the type of failure, but for the ones present in this work, their use increased fault detection performance. Also, the use of the gearbox oil temperature completely eliminated the fault detection capabilities of the model. The nacelle temperature also seemed to decrease the fault detection performance of the model. This means that although simultaneity features can improve the temperature modelling performance of the model, they decrease the fault detection performance.These results are aligned with the works in the literature that showed that highly correlated features, such as gearbox oil temperature, resulted in lower fault detection performance. But this work also showed that the origin of the problem is not the features being highly correlated, but due to the simultaneity nature of their causal relation

Finally, this work also showed that besides using autoregressive features to improve the fault detection performance of causal models, one can also use post-processing techniques on the residuals. The results showed that the active power filter considerably increased the fault detection performance of the model, due to filtering out the regimes that this model didn't learn as well. Indeed, this is an area for further research. Namely, techniques to handle imbalanced datasets, such as under-sampling, may be investigated. Besides that, more post-processing techniques based in other input features, such as the pitch angle, may result in better filtering of the regimes not learned by the causal model and thus increase fault detection performance.

References

[1] International Energy Agency. *Global Energy & CO2 Status Report*. International Energy Agency, 2018.

[2] DNV GL. *Energy Transition Outlook*. DNV GL, 2018.

[3] EWEA. *The Economics of Wind Energy*, 2009.

[4] Kevin Leahy. *Data analytics for fault prediction and diagnosis in wind turbines*. PhD thesis, University College Cork, 2018.

[5] IRENA. *The Power to Change: Solar and Wind Cost Reduction Potential to 2025*, 2016.

[6] A Zaher, Stephen McArthur, David Infield, and Y Patel. Online wind turbine fault detection through automated scada data analysis. *Wind Energy*, 2009.

[7] R Mesquita, José Carvalho, and F Pires. Neural networks for condition monitoring of wind turbines gearbox. *J. Energy Power Eng.*, 2012.

[8] R. F. Mesquita Brandão, J. A. Beleza Carvalho, and F. P. Maciel Barbosa. Intelligent system for fault detection in wind turbines gearbox. In *2015 IEEE Eindhoven PowerTech*, 2015.

[9] Meik Schlechtingen and Ilmar Santos. Comparative analysis of neural network and regression based condition monitoring approaches for wind turbine fault detection. *Mechanical Systems and Signal Processing*, 2011.

[10] Martin Bach-Andersen, Bo Rømer-Odgaard, and Ole Winther. Flexible non-linear predictive models for large-scale wind turbine diagnostics. *Wind Energy*, 2016.

[11] Daniel Karlsson. *Wind Turbine Performance Monitoring using Artificial Neural Networks*

With a Multi-Dimensional Data Filtering Approach. PhD thesis, Chalmers University of Technology, 2014.

[12] Yue Cui, Pramod Bangalore, and Lina Bertling Tjernberg. An anomaly detection approach based on machine learning and scada data for condition monitoring of wind turbines. In *2018 IEEE International Conference on Probabilistic Methods Applied to Power Systems (PMAPS)*, 2018.

[13] Pramod Bangalore, S Letzgus, Daniel Karlsson, and Michael Patriksson. An artificial neural network based condition monitoring method for wind turbines, with application to the monitoring of the gearbox. *Wind Energy*, 2017.

[14] Jannis Tautz-Weinert. *Improved wind turbine monitoring using operational data.* PhD thesis, Loughborough University, 2018.

[15] Martin Bach-Andersen. *A Diagnostic and Predictive Framework for Wind Turbine Drive Train Monitoring.* PhD thesis, Technical University of Denmark, 2017.

[16] L Colone, M Reder, N Dimitrov, and D Straub. Assessing the utility of early warning systems for detecting failures in major wind turbine components. *Journal of Physics: Conference Series*, 2018.

[17] Yingying Zhao, Dongsheng Li, Ao Dong, Dahai Kang, Qin Lv, and Li Shang. Fault prediction and diagnosis of wind turbine generators using scada data. *Energies*, 2017.

[18] Christopher Krauss, Xuan Anh Do, and Nicolas Huck. Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the sp 500. *European Journal of Operational Research*, 2016.

[19] Souhaib Ben Taieb and Rob Hyndman. A gradient boosting approach to the kaggle load forecasting competition. *International Journal of Forecasting*, 2013.

[20] James Robert Lloyd. Gefcom2012 hierarchical load forecasting: Gradient boosting machines and gaussian processes. *International Journal of Forecasting*, 2013.

[21] Wes Mckinney. Data structures for statistical computing in python. *Proceedings of the 9th Python in Science Conference*, 2010.

[22] Plotly Technologies Inc. Collaborative data science, 2015.

[23] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems 30.* 2017.

[24] James J. Heckman. Econometric causality. *International Statistical Review*, 2008.

[25] Judea Pearl. An introduction to causal inference. *The international journal of biostatistics*, 2010.

[26] Matthew Rocklin. Dask: Parallel computation with blocked algorithms and task scheduling. In *SCIPY 2015*, 2015.

[27] Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. 2006.

[28] J Tautz-Weinert and S J Watson. Comparison of different modelling approaches of drive train temperature for the purposes of wind turbine failure detection. *Journal of Physics: Conference Series*, 2016.

[29] A Kusiak and Anoop Verma. Analyzing bearing faults in wind turbines: A data-mining approach. *Renewable Energy*, 2012.

[30] Meik Schlechtingen, Ilmar Santos, and Sofiane Achiche. Wind turbine condition monitoring based on scada data using normal behavior models. part 1: System description. *Applied Soft Computing*, 2013.