Diagnostic Models for Wind Turbine Gearbox Components Using SCADA Time Series Data

Preprint

Rafael Orozco
Bucknell University

Shuangwen Sheng and Caleb Phillips
National Renewable Energy Laboratory

Presented at the 2018 IEEE International Conference on Prognostics and Health Management
Seattle, Washington
June 11–13, 2018

Suggested Citation
Orozco, Rafael, Shuangwen Sheng, and Caleb Phillips. 2018. "Diagnostic Models for Wind Turbine Gearbox Components Using SCADA Time Series Data: Preprint." Golden, CO: National Renewable Energy Laboratory. NREL/CP-5000-71166. https://www.nrel.gov/docs/fy18osti/71166.

NREL is a national laboratory of the U.S. Department of Energy
Office of Energy Efficiency & Renewable Energy
Operated by the Alliance for Sustainable Energy, LLC

This report is available at no cost from the National Renewable Energy Laboratory (NREL) at www.nrel.gov/publications.

Conference Paper
NREL/CP-5000-71166
July 2018

Contract No. DE-AC36-08GO28308

NOTICE

This work was authored by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding provided by the U.S. Department of Energy Office of Energy Efficiency and Renewable Energy Wind Energy Technologies Office. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

Cover Photos by Dennis Schroeder: (left to right) NREL 26173, NREL 18302, NREL 19758, NREL 29642, NREL 19795.

NREL prints on paper that contains recycled content.

Diagnostic Models for Wind Turbine Gearbox Components Using SCADA Time Series Data

Rafael Orozco
Computer Science
Bucknell University
Lewisburg, Pennsylvania
rao010@bucknell.edu

Shuangwen Sheng
National Wind Technology Center
National Renewable Energy Laboratory
Golden, Colorado, USA
Shuangwen.Sheng@nrel.gov

Caleb Phillips
Computational Sciences
National Renewable Energy Laboratory
Golden, Colorado, USA
Caleb.Phillips@nrel.gov

***Abstract*—This paper presents a scalable, autonomous framework for diagnostics of wind turbine gearbox failures using a multi-feature supervisory control and data acquisition data set spanning multiple years. Because of the size of the data set studied, all algorithms were constructed to be scalable using a Spark-Hadoop data framework. An unsupervised approach was used to detect significant failures based on predetermined criteria. The initial criteria selected were an abnormal spike in turbine component temperature followed by a turbine power off, which helps reduce the number of potential false alarms. To detect abnormal spikes in component temperature, a model was introduced to adjust the temperature data for effects caused by ambient temperature and normal temperature increases when loaded. This study evaluates methods for normalizing temperature sensor data to identify anomalies. The models that performed the best were a linear regression and a multivariate polynomial regression. The proposed process for finding failures has tunable parameters that can be adjusted to be more or less sensitive. The combination of sensor data normalization and application of these criteria is successful in finding turbine failures resulting in downtime. The proposed methods can operate without failure or maintenance logs and can be utilized for offline analysis of large high-resolution data sets.**

Keywords: diagnostic model; machine learning; big data; wind turbine; gearbox; component failures

I. INTRODUCTION

One mission of the U.S. Department of Energy (DOE) is to promote the adoption of clean energy sources. Wind energy will play a big role in this energy revolution. DOE has published scenarios that predict 20% of U.S. energy needs coming from wind by 2030 [1]. To further promote the adoption of wind energy, it is necessary to make wind turbines more resilient to failures. This can be addressed in a multifaceted approach, such as design, testing, and operation and maintenance (O&M) [2]. By increasing the reliability of turbines, substantial gains can be made to minimize O&M costs of wind energy.

Once turbines are installed in a wind power plant, the main opportunity to improve turbine availability lies in improving O&M practices. Performance monitoring based on turbine supervisory control and data acquisition (SCADA) data and condition monitoring based on dedicated instrumentation, such as vibration and oil analysis, have been actively explored by the wind industry [3] to help accomplish this. The advantage with performance monitoring is SCADA data is readily available and does not need additional investments for hardware as condition monitoring typically does. On the other hand, there are still opportunities for mining of SCADA data by the wind industry to help improve turbine availability and reduce O&M costs. This study extends existing work by National Renewable Energy Laboratory (NREL) researchers on wind turbine gearboxes [4], which have been shown to be the most costly subsystem to maintain throughout a turbine's 20-year design life, by using data-driven approaches to catalog and understand component failures in wind turbine gearboxes.

This work uses historical data of wind turbines to detect failures identified by overheating of components inside turbine gearboxes. The data set is 948 gigabytes (GB), so it requires scalable algorithms and use of NREL's high-performance computing (HPC) resources. The data include many features (e.g., multiple component temperatures, power generated, wind velocity) over a long period of time. We developed a model that adjusts component temperature for weather and power production returning a normalized temperature value. This computational model demonstrates temperature behavior under normal conditions. By comparing the model predictions with the observed data, we can see where the turbine is behaving in abnormal ways. These anomalies in the data point to possible failures in the components of the turbine. The diagnostic framework developed identifies failures as a combination of substantial temperature anomalies followed by system downtime.

In Section II (Background), we will introduce the data set used for this study and discuss prior related work to identify turbine failures from operational data. In Section III (Methods), we introduce a new method for normalization of SCADA data channels and a diagnostic algorithm to identify gearbox failures. In Section IV (Results), we apply the proposed method to the data and present the results. Finally, in Section V (Conclusion) we summarize the contribution and areas for future work.

II. Background

In this project, we work with the Continuous Reliability Enhancement for Wind (CREW) data set. This is a large data set of SCADA data compiled by Sandia National Laboratories, with the mission of characterizing reliability performance issues and identifying opportunities for improving reliability and availability performance of the U.S. national wind energy infrastructure [5]. The version of CREW we use includes data from 614 turbines from 7 different plants, providing 388 years of turbine data as shown in Table I.

Recording SCADA data from wind turbines is an industry standard practice and typically includes both 10-minute average time series data and turbine status codes. The information available in SCADA is helpful to analyze turbine status in a general manner but there are many SCADA data sets that only have time series channels and not failure status codes or supplementary maintenance logs, which might be used to identify or diagnose failures. This makes the time series SCADA data less useful for machine learning training.

TABLE I. CREW PLANT STATISTICS

Wind Plant ID	**Turbine Numbers**	**Turbine Days**	**Plant Rated Power (MW)**	**Turbine Rated Power (MW)**	**Native Resolution (sec)**
1	41	20,902	61.5	1.5	5
2	147	22,653	207.5	1.5, 1.6	2
3	69	25,863	103.5	1.5	2
4	102	39,863	153	1.5	2
5	53	5,291	108.65	2.05	7 - 8
6	66	20,832	132	2	2
7	136	6,259	204	1.5	5 - 6
All Plants	614	141,666	970.15	1.5, 1.6, 2, 2.05	2 - 8

Various efforts have been made for mining wind power plant SCADA data to support turbine component health diagnostics. Most of these efforts first develop a model for normal conditions when the monitored turbine is considered healthy, and then analysts compare measured data against predictions given by the developed model to evaluate the deviations, which are used in turbine component fault diagnostics. In [6], an online wind turbine fault detection framework was presented and it integrated diagnostics models targeting various turbine subsystems, such as the gearbox and generator. By examining the deviations of test data sets from models developed under normal conditions, gearbox failures, based on cooling oil and bearing temperature data, and generator failures, based on generator winding temperature, were successfully detected. In [7], a damage model was developed by targeting one failure mode to describe the relationships between turbine operating environment, applied loads, and damage accumulation rates. Using measured SCADA data and the developed damage model, a reliability value for the monitored gearbox to fail under the targeted failure mode could be obtained. In [8], a nonparametric regression method named least squares support vector regression was proposed to characterize the baseline relationship between turbine responses and wind resources. Based on the baseline model, response and residual control charts were derived and used to identify abnormal turbine responses including diagnostics of a wind turbine gearbox failure. Most of these efforts used relatively small data sets, and their developed baseline models are not scalable or hard to generalize.

The method we propose is an unsupervised approach of labeling failures in SCADA data sets such as CREW. Unsupervised learning is a class of machine learning that operates without requiring data to be labeled. These algorithms identify structure in data including patterns, relationships, and groups. Using unsupervised analysis can unlock value in these data sets for identifying, diagnosing, and summarizing failures as well as providing input to prognostic models (i.e., supervised machine learning) that may avoid failures by detecting them before they occur.

The CREW data set includes many features. The failure detection method that we present makes use of component temperature readings and power output. Using only two features to train a model is a novel way of performing diagnostic analysis. Normally, models make use of more features, such as the 18 features presented in the work of Kusiak et al. [9]. The method we suggest is quicker and requires fewer features. Therefore, it can be scaled more easily to larger data sets and is more generally applicable because temperature and power data are prevalent in most SCADA data sets.

III. Methods and Process

A. *Data Preprocessing*

The CREW data set is loaded initially as a large CSV data file that was extracted from a Microsoft SQL database. In order to perform scalable analysis on this data, we first process it into a more manageable format and size. We down sampled the data by taking the moving average of all the time-series features. The resolution selected was 300 measurements per window, corresponding to a 10-minute windowed average for data at 2 second native resolution (plants 2, 3, 4 & 6). This 10-minute average is an industry standard for summarized SCADA streams and is similar to what many operators may archive. Plants 1, 5, and 7 provide data at a lower native resolution, so the resulting temporal resolution is larger.

The data include turbines from multiple wind power plants and were not standardized. This presented a challenge because data from one plant (Plant 6 in Table I) did not have the temperature readings required for our full failure analysis. As a result, we chose to leave this plant out of the final results.

B. *Proposed Failure Detection Process*

The process to detect failures involves four steps. First, we train a model that maps the two independent variables, ambient temperature and power output, to the dependent variable, component temperature. Next, data output from the trained model is used to create a residual between the modeled output and the observed data. These data points show when the temperature is abnormally high. From these data points that show abnormally high temperatures, we filter out only those that are followed by a turbine shutdown. In this case, we define a turbine shutdown to be power output being zero when wind velocity data shows that the wind is still blowing. Lastly, we can nominate these filtered-out data points as a component failure. After flagging all failures for a turbine, we can then perform plantwide failure analysis.

C. *Selecting Process Parameters*

The proposed process gives rise to several adjustable parameters:

- A temperature threshold that is denominated as a temperature spike: we select the 99th quantile of the model residue to be the threshold.
- Amount of turbine components that share a temperature spike: we choose that at least half of the turbine components must “agree” that there is a temperature spike.
- Number of data points in the future after a temperature spike to look out for turbine power-off time: this is a sensitivity parameter. Refer to the second column of Table II for chosen values.
- Amount of turbine power-off time that will be declared a power off: this is a sensitivity parameter. Refer to the third column of Table II for chosen values.

A study presented by Reder et al., which studied more than 4,300 turbines, concluded that geared wind turbines rated at 1 MW or more fail at the rate of 0.52 per year [10]. The sensitivity parameters of the process were adjusted by hand using this statistic as a guide.

TABLE II. FINAL SENSITIVITY PARAMETERS

Wind Plant ID	Data Points in Future Lookout	Downtime (%)
1	120	99
2	5	60
3	15	80
4	60	90
5	120	90
6	NA	NA
7	10	80

Table II shows the final sensitivity parameters that were chosen. As a rule of thumb, lower values of these parameters will make the process criteria less “strict,” resulting in more detected failures. Areas where parameters may be auto-tuned to the point of only catching true failures is a topic for future work.

D. Methods for Modeling Normal Temperature Behavior

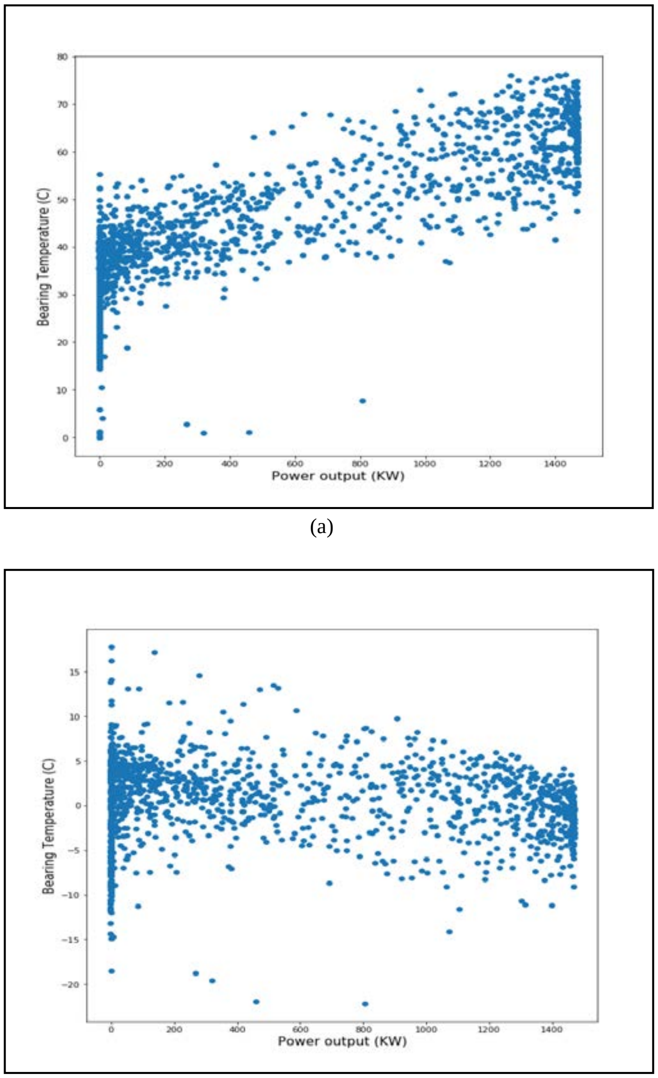
To classify temperature data as abnormal, we first adjust the data to eliminate any effects that are caused by the ambient temperature and normal temperature increases when the turbine is generating power. The motivation is that we want to avoid flagging high temperatures that are not abnormal—for example, a spike in temperature caused by a hot day (adjusting for ambient temperature), or a spike caused by the turbine working harder (adjusting for power output). These two examples are expected behavior and do not constitute a failure in the turbine.

To adjust for the ambient data and the power output, we will construct a model, as shown in (1), that learns what the causal relationship is between the input features (ambient temperature and power output) and the output features (component temperatures). Once we have this model, we will subtract the model output temperature data from the raw temperature data. This is called an adjusted temperature and has been used to detect abnormal behavior in sensor readings [9].

$$T_a(i) = T_r(i) - T_m(i)\{T_e(i), P(i)\} \tag{1}$$

where $\mathrm{T_a}$ is adjusted temperature, $\mathrm{T_r}$ is raw temperature, $\mathrm{T_m}$ is modeled temperature, $\mathrm{T_e}$ is environmental or ambient temperature, and P is power. There are many models that can be used for this purpose. To find the best model for the task, we selected several candidates that were tested according to their success with three metrics: the root-mean-square error (RMSE), the Pearson correlation coefficient (PCC), and the Shapiro-Wilk normality test (SWNT).

Conventionally, the fit of a model is measured with the RMSE. We added the PCC and SWNT because these two metrics can measure how much of the effect ambient temperature and power output has been eliminated from the raw data. Visually, this will look like the scatter plot of the independent variable (ambient temperature or power output) than the dependent variable (component temperature that we are trying to model), changing from showing a clear



(a)

(b)

Figure 1. Raw temperature correlated with power (a) and adjusted temperature based on a trained model minimized such a correlation to strengthen impacts from component failures (b).

correlation to looking more like two random variables (Fig. 1). We also explored the use of a quadrat test for total spatial randomness but found that it did not contribute meaningfully to our model selection.

The RMSE is used to measure how close of a fit the proposed model is in comparison to the observed data. The best model will return a smaller RMSE. The RMSE is defined as:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(c_i - \overline{c_i})^2} \tag{2}$$

where N is the number of observations, c is observed data, and $\overline{c}$ is the modeled observation.

The Pearson correlation coefficient (PCC) will be applied on the model-adjusted data to track how much of the effect ambient temperature and power has been eliminated from the raw data. The PCC lets us quantify this "correlation elimination." The best model will return a lower PCC. The Pearson correlation coefficient, ρ, between two variables, X and Y, is defined as:

$$\rho(X,Y) = \frac{Cov(X,Y)}{\sqrt{Var(X)Var(Y)}} \tag{3}$$

The Shapiro-Wilk normality test will be another metric used to quantify how much of the input data correlation is eliminated by the model. A normality test shows how much a certain sample differs from a normal distribution. The normality metric of a model residual can be used to measure the fit of the model [11]. This is because if the model is a good fit, then the residual should be close to a normal distribution. The Shapiro-Wilk number (W) is the result of the formula:

$$W = \frac{(\sum_{i=1}^{n} a_i x_{(i)})^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{4}$$

In our testing, we used the statistic W' = 1 - W so that the better model returns a lower number.

The models we tested in this project are described in the following subsections:

1) *Linear regression.* If we presume that there is a linear relationship between the input and output variables, then we can define a linear model of the form:

$$T_m(i) = \alpha_0 + \alpha_1 * T_e(i) + \alpha_2 * P(i) \tag{5}$$

where T_m is modeled temperature, T_e is environmental temperature, P is power, and α_0, α_1, and α_2 are linear regression coefficients. We used the linear model package from Scikit-learn [16]. This package finds the coefficients that minimize the residual sum of squares of the training data.

2) *Multivariate polynomial regression.* This type of regression is useful because it can be treated as a linear regression, wherein the linear variables are the coefficients of the polynomial expression:

$$T_m(i) = \alpha_0 + \alpha_1 * T_e^2(i) + \alpha_2 * P(i) * T_e(i) + \alpha_3 * P^2(i) \tag{6}$$

where T_m is modeled temperature, T_e is environmental temperature, P is power, and α_0, α_1, α_2, and α_3 are polynomial regression coefficients.

3) *Random forest.* Conventionally, the random forest algorithm is used as a classifier. For our purposes, it works well as a regression. This is because it "classifies" the input variables to certain output variables, which is in essence a form of regression. The random forest algorithm has been shown to be a very robust form of regression [12].

To train a random forest regression model, we used the machine learning library of spark, MLlib [17]. A random forest of 16 trees was used. Increasing the number of trees after this number did not show a substantial increase in model performance.

4) *Neural network.* Neural networks have been used many times for wind turbine modeling [9]. They are used because they are robust and good at modeling nonlinear behaviors. Similar to the random forest algorithm, a neural network is a "black box," in which the resulting model cannot be easily interpreted. For this project, we used the neural network implementation from Scikit-learn [16], which is named a multilayer perceptron. We used a 2-18-1 structure to model component temperature given the two inputs: ambient temperature and power output. We selected this structure by taking into consideration previous work in neural network modeling for turbine components [10].

E. *Processing Data at Scale*

Because the CREW data set is large (948 GB), all parts of the process and algorithms in this project were coded in a scalable way. The main library used was Spark because it allows the algorithms to be run on a local or HPC system. All code was run on the NREL HPC Sparkplug cluster environment, which implements the Hortonworks Data Platform [13] within Openstack [14]. The specifications on the cluster are: 5 cores per executor and 20 executor nodes with 23 GB of memory each.

In order to train the different models, we used a random sample of the data set. The model was given 5000 data points to train itself. Some turbines had fewer data points than others, so this sample quantity was selected because it would permit the greatest number of turbines to be analyzed.

One cycle of the algorithm includes the following analysis for one turbine:

- Down-sample temperature and power data to 5000 points
- Use sample data to train model for each temperature reading
- Use modeled temperature data to adjust the entire temperature data
- Flag an adjusted temperature when it is over the 99th quantile
- Flag data points in which at least half of the temperature channels agree that there is overheating
- Filter out temperature flags by picking the ones that are followed by a turbine shutdown
- Label the data points corresponding to these filtered temperature flags as turbine failures. Note: a chain of data points labeled as failures will be counted as one failure.

The time elapsed ranged from 11 s to 17 s per turbine, depending on the number of data points available for the turbine.

IV. Results

A. *Model Performance*

TABLE III. Final Sensitivity Parameters

Model	RMSE	Pearson (Ambient/Power)	Normality (Ambient/Power)
Linear Regression	8.58	3.27e-16/4.63e-16	0.96/0.97
Multivariate Polynomial Regression	7.66	4.64e-06/4.64e-06	0.94/0.98
Random Forest	9.47	0.56/0.01	0.94/0.96
Neural Network	10.36	0.47/0.14	0.93/0.97

We evaluated all four models using the three metrics (Table III). Each turbine had eight temperature readings so we modeled each one and averaged the metric results of all eight models. Because the PCC and normality metric track two different variables (ambient temperature and power output,) we show the results of each variable separately. This reveals that both the random forest and the neural network have a considerably higher PCC score when eliminating ambient correlation in comparison with eliminating power correlation. This result makes sense because we expect that a more complex relationship exists between component temperature and power output as a result of various speed turbine operation.

Each of these metrics will show a lower score if the model is performing better. For the purposes of our process, we are mostly interested in a low Pearson score. The lowest Pearson scores are found in the linear regression model. Considering the complex nonlinear behavior of the turbine, it is noteworthy that the linear model also performed well on all metrics.

B. *Adjusting Temperature*

From the temperature values (Fig. 2), we can see that the adjusted temperature shows a spike, wherein the unadjusted temperature does not. This was a spike that was “hidden” before the data was adjusted for power and ambient temperature. Adjusting the temperature also fulfills the purpose of removing false positives. A false positive would be a temperature spike that can be explained by a high ambient temperature or a wind turbine that is producing more energy. Once such a spike is found, the data point is labeled as a failure if it is followed by a turbine shutdown.

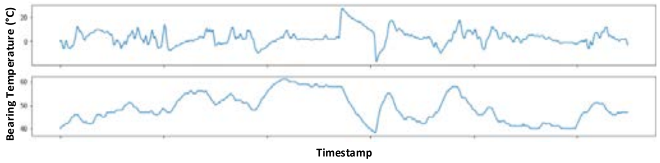


Figure 2. Temperature data adjusted for power output by the trained model (top) and corresponding raw temperature data (bottom).

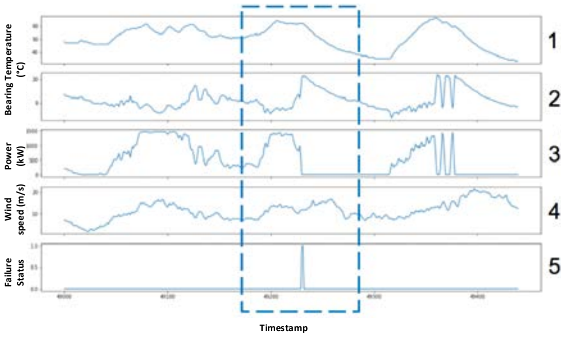


Figure 3. These five graphs demonstrate the full process by which our method detects a failure (highlighted by the dashed box).

C. *Detected Failures*

Following the graphs shown in Fig. 3, this is how our diagnostic process labels failures:

1. Raw temperature data. The unadjusted temperature for this component (Bearing A) does not show any high temperature spikes.
2. Model adjusted data. The adjusted temperature shows a temperature spike, meaning that according to the model the turbine is hotter than it should be given the ambient temperature and power output conditions.
3. Power output data. Drops to zero values revealing a turbine shutdown.
4. Wind speed data. Wind is still blowing, so the turbine shutdown is not caused by low wind speeds.
5. Failure label. Timestamp is labeled as turbine failure by diagnostic process.

D. *Failure Detection Statistics on All Plants*

Lastly, we have also calculated failure detection statistics for all six power plants (Table IV). Each plant used the same diagnostic process but with different parameters (Table II). We can see from the table that the average failure rate over all the plants (0.42) is very close to the failure rate found in actual turbines of the same type (0.52). Considering the first 30 turbines of each power plant, we made the following representative heat map showing failure rates by turbine (Fig. 4). Each cell represents one turbine labeled with the number of failures detected. Each row represents the turbines from a single plant. Turbines 16 and 29 in plants 1 and 5 showed the highest rate of failure, whereas some turbines did not appear to fail within the data set studied.

TABLE IV. PLANTWIDE FAILURE STATISTICS

Wind Plant ID	Turbines	Failures Detected	Failures/ Turbine/ Year
1	41	78	0.63
2	147	25	0.17
3	69	56	0.37
4	102	69	0.46
5	53	91	0.60
6	NA	NA	NA
7	136	46	0.31
All Plants	614	365	0.42

As is typical at wind plants, the data analyzed here were not provided with failure or maintenance logs. The conservative method described here appears capable of identifying failures. However, a full validation with high-resolution sensor data paired with digitized maintenance logs will be a topic of future work.

V. CONCLUSION

We have shown that temperature and power data alone can be used to identify potentially catastrophic failures using SCADA data. We also found that, with the metrics chosen, the linear regression model and the fourth-order polynomial model regression had the greatest ability to reliably normalize the temperature data. Although both models can be applied at scale, a linear versus a polynomial regression model is preferred for this task because of its simplicity and efficiency.

The parameters used in the process were adjustable to be consistent with published failure statistics [10]. According to the paper, turbines rated at 1 MW or higher experience 0.52 failures per year on average. Our process applied on the CREW set detected 0.42 failures per year.

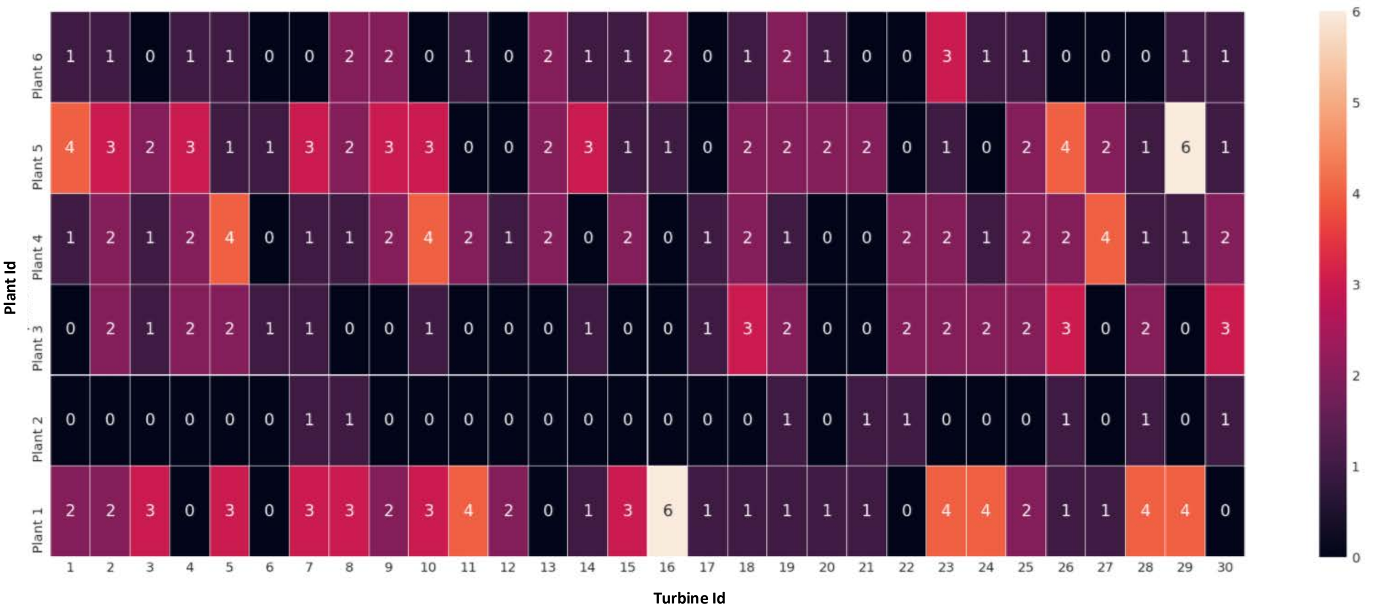


Figure 4. Number of failures identified for first thirty turbines at all six plants (highest on turbines 16 at plant 1 and 29 at plant 5)

Scalable diagnostic models such as these can be used to mine historical data sets for training machine-learning prognostic models, thereby providing a rich historical training set even in the absence of hand-labeled data, which may rarely be available. This is a capability that will be increasingly important as operational data from power plants becomes increasingly large and complex due to increased numbers of devices and sensors, and increasing resolution of data due to increasing prevalence of condition monitoring systems [15].

We have identified several opportune areas for additional work. In future work, we will use models with more input parameters and will explore the sensitivity of results to the native data resolution and method of down-sampling Other authors have shown success using models with up to 18 input variables [9], which suggests that failures may be identified with greater accuracy using more data.

Acknowledgment

This work was supported in part by the Office of Science and Office of Workforce Development for Teachers and Scientists under the Science Undergraduate Laboratory Internships Program.

The Alliance for Sustainable Energy, LLC (Alliance) is the manager and operator of the National Renewable Energy Laboratory (NREL). NREL is a national laboratory of the U.S. Department of Energy, Office of Energy Efficiency and Renewable Energy. This work was authored by the Alliance and supported by the U. S. Department of Energy under Contract No. DE-AC36-08GO28308. Funding was provided by the U.S. Department of Energy Office of Energy Efficiency and Renewable Energy Wind Energy Technologies Office. The views expressed in the article do not necessarily represent the views of the U.S. Department of Energy or the U.S. government. The U.S. government retains, and the publisher, by accepting the article for publication, acknowledges that the U.S. government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. government purposes.

References

[1] S. Lindenberg, B. Smith and K. O'Dell, "20% Wind Energy by 2030," U.S. Department of Energy, Renewable Energy Consulting Services, Energetics Incorporated, Washington, D.C., 2008.

[2] S. Sheng and R. O'Connor, "Reliability of Wind Turbines," in *Wind Energy Engineering: A Handbook for Onshore and Offshore Wind Turbines*, London, UK, Academic Press, 2017, pp. 299-327.

[3] S. Sheng, "Improving Component Reliability Through Performance and Condition Monitoring Data Analysis," 2015. [Online]. Available: https://www.nrel.gov/docs/fy15osti/64027.pdf.

[4] S. Sheng, C. Phillips and N. Wunder, "Gearbox Reliability Database," National Renewable Energy Laboratory (NREL), 19 January 2018. [Online]. Available: http://grd.nrel.gov.

[5] V. Peters, A. Ogilvie and C. Bond, "CREW Database: Wind Plant Reliability Benchmark," Sandia National Laboratories, Albuquerque, NM, 2012.

[6] A. Zaher, S. D. J. McArthur, D. G. Infield and Y. Patel, "Online wind turbine fault detection through automated SCADA data analysis," *Wind Energy,* vol. 12, no. 6, pp. 574-593, 2009.

[7] C. S. Gray and S. J. Watson, "Physics of failure approach to wind turbine condition based maintenance," *Wind Energy,* vol. 13, no. 5, pp. 395-405, 2010.

[8] N. Yampikulsakul, B. Eunshin, H. Shuai, S. Shuangwen and Y. Mingdi, "Condition Monitoring of Wind Power System With Nonparametric Regression Analysis," *Energy Conversion, IEEE Transactions on,* vol. 29, no. 2, pp. 288-299, 2014.

[9] A. Kusiak and A. Verma, "Analyzing bearing faults in wind turbines: A data-mining approach," *Renewable Energy,* vol. 48, pp. 110-116, 2012.

[10] R. M. D., E. Gonzalez and J. Malero, "Wind Turbine Failures - Tackling Current Problems in Failure Data Analysis," *Journal of Physics: Conference Series,* vol. 753, no. 7, 2016.

[11] P. Sudhir R. and X. Zhang, "Testing for normality in linear regression models," *Journal of Statistical Computation and Simulation,* vol. 80, no. 10, pp. 1101-1113, 2010.

[12] A. Liaw and M. Wiener, "Classification and regression by randomForest," *R News,* pp. 18-22, 2002.

[13] Hortonworks, "Open Source Solutions for Data-at-Rest," [Online]. Available: http://hortonworks.com.

[14] S. Omar, M. Aissaoui and M. Eleuldj, "Openstack: toward an open-source solution for cloud computing," *International Journal of Computer Applications,* vol. 55, no. 3, 2012.

[15] W. Yang, P. J. Tavner, C. J. Crabtree, Y. Feng and Y. Qiu, "Wind turbine condition monitoring: technical and commercial challenges," *Wind Energy,* vol. 17, no. 5, pp. 673-693, 2014.

[16] F. Pedregosa and e. al., "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research,* vol. 12, pp. 2825-2830, Oct 2011.

[17] Apache Software Foundation, "MLLib - Apache Spark's Scalable Machine Learning Library," June 2017. [Online]. Available: http://spark.apache.org/mllib/.