Multi-task distribution learning approach to anomaly detection of operational states of wind turbines

Stephan Vogt\textsuperscript{1}, Scott Otterson\textsuperscript{1,2}, Volker Berkhout\textsuperscript{1}

\textsuperscript{1}Fraunhofer Institute for Energy System Technology and Energy Economy (IEE), Königstor 59, 34119 Kassel, Germany
\textsuperscript{2}TenneT TSO GmbH, Bernecker Straße 70, 95448 Bayreuth, Germany

stephan.vogt@iee.fraunhofer.de

Abstract. The detection of abnormal operation modes is of fundamental importance for both operational management and predictive maintenance of wind turbines. Anomaly detection approaches in this context should consider the additional information content that probabilistic models can provide. Instead of binary anomaly classification, the probabilistic information is necessary for proper decision making and risk assessment. Common models, such as quantile and distribution regression can provide probabilistic information. While they are appropriate in predicting the cumulative distribution function, they struggle to accurately describe the probability of an event to occur. In this article we present a new, multi-task learning based approach for a continuous distribution regression with deep neural networks. Using real-world data from an offshore wind turbine, we show that with this model we can better reflect the probability of observed events than with conventional methods. While the predicted cumulative distribution function has a similar quality and no significant differences are visible in the continuous ranked probability score, the probability density function will be substantially smoother. This is also reflected in a significantly lower ignorance score.

1. Introduction
Improving the performance of wind turbines by applying predictive maintenance techniques with data-driven approaches and machine learning is a field with very active ongoing research and high potential for increased yield and reduced operational cost. A key challenge for machine learning algorithms such as artificial neural networks is to detect time periods where a system, such as a wind turbine operates in an abnormal state. Methods for this anomaly detection problem in wind energy have been discussed for supervised \cite{1-3} and semi- or unsupervised approaches \cite{4,5}. Moreover, the assessment of a possibly detected anomaly which may depend on wide range of features within and outside the turbine should rather be expressed in a probabilistic way instead of giving a binary classification. This allows adjusting for the level of confidence with which an anomaly is detected and helps to reduce the number of false positives which may lead to unnecessary service and inspection efforts and low acceptance of a predictive maintenance system. Approaches to use probabilistic failure information in turbine maintenance have been described by Bangalore and Patriksson \cite{6}.
The paper will start with a description of the basic characteristics of artificial neural networks predicting the turbine state based on measured sensor values. The quantile and distribution regression approach will be described and shortcomings will be pointed out.

Then we present the concept of multi-task learning in the application for modeling uncertainties with conditional distributions. Based on this concept we develop a new approach and show improvements compared to quantile and distribution regression algorithms. Finally the paper will show the results of the newly developed algorithm when applied on a real field data set from an offshore wind turbine.

An intuitive and widely used technique for anomaly-detection in measured systems is to predict every sensor based on all remaining sensors. Most sensors show values that can be plausibilized using other sensors. Due to uncertainty in the measurements and because the state of the turbine can only be partially observed, each sensor should be modelled probabilistically. The estimated conditional distribution can then be used to quantify the plausibility of a sensor value given the state of the turbine with respect to other sensor measurements. Widespread state-of-the-art probabilistic models are quantile regression (QR) and distribution regression (DR). Both models can be combined with artificial neural networks to better represent non-linear behavior. The following shows how QR and DR can be applied to neural networks and how neural networks can be combined even more efficiently with distribution regression in a multi-task learning setup.

2. Model Configurations for Nonparametric Conditional Distributions with Neural Networks

Modern artificial neural networks (ANN) are composed by a set of partially differentiable operations, input variables and parameter matrices and vectors. The key concept is to predict values based on the operational data from the training period by means of advanced statistical regression.

Depending on the demanded task, a suitable loss function is used to evaluate the trained parameters to create an indicator for the quality of the prediction by the ANN. The goal here is for the regression models to represent the conditional distribution in the form of the cumulative distribution function (CDF) as a function of independent variables, such as other sensors. Any combination of independent sensors based on expert knowledge can be selected to estimate the conditional distribution of a different dependent sensor for each time step. By comparing the distribution with the actual observation of the dependent sensor, it can be evaluated whether the respective sensor shows anomalous behavior. In the experiment the model approach of quantile regression, distribution regression and the multi-task learning distribution regression, which is presented in this paper, were investigated in combination with an artificial neural network. The following paragraphs describe the configurations used and illustrate them in Figure 1.
Quantile regression (QR):

Distribution regression (DR):

Multi-Task learning distribution regression (MTL):

Figure 1. Schematic model structure for the quantile regression neural network (QRNN / top), the distribution regression neural network (DRNN / middle) and the multi-task learning distribution regression (mtlDRNN / bottom) with their respective discretization of the cumulative distribution probability function.

2.1. Quantile Regression Neural Network QRNN

The neural network in combination with quantile regression was first described by Taylor in 2000 [9]. The already known method of quantile regression, described by Koenker in 1978 [10], was applied and the gradient $\nabla L_\tau$ of the quantile loss function

$$L_\tau = \begin{cases} \tau \cdot |y - \hat{y}_\tau| & \text{if } y \geq \hat{y}_\tau \\ (1 - \tau) \cdot |y - \hat{y}_\tau| & \text{else} \end{cases}$$

where $y$ is the observed power and $\hat{y}_\tau$ is the predicted value of the quantile $\tau$, was used to tune the model parameters of the network. Ideally, with a well-trained model $P(Y \leq \hat{y}_\tau | x) = \tau$ should apply, where $Y$ is the target random variable and $x$ is a vector of the independent sensors values. A further development and description of the corresponding toolbox in R has been provided by Cannon [11].

The quantile regression neural network (QRNN) used in this experiment has the following structure: An output unit was added to the neural network for each quantile between $\tau_1 = 0.05$ and $\tau_n = 0.95$ in 0.05 steps. For this purpose, a linear function is selected as an activation function, since the quantile predictions could be positive as well as negative real-valued values. Each output unit is trained with the same target value, but not with the same loss function, which is the quantile score described by Koenker [10]. It evaluates deviations upwards or downwards differently depending on the quantile. The gradients of the loss function are computed (backpropagation) and used to optimize the neural network weights with gradient descent. While the weights of the respective output unit only have to minimize the individual loss function per quantile, the weights of previous layers must be adjusted to optimize all loss functions at the same time, since they are shared by all output units.

2.2. Distribution Regression Neural Network DRNN

A description of the distribution regression by Peracchi can be found in [12]. Similar to the QRNN, the distribution regression neural network (DRNN) has several output units in the distribution regression. In contrast to QRNN, there is only one loss function. However, the target value differs from unit to unit. Instead of a real-valued target value, it is transformed with several threshold values into a higher-dimensional categorical target value. More precisely, 100 uniform thresholds are chosen between the slightly less than the smallest and slightly more than the largest target value, so that the
thresholds cover 110% of the range of the target value. For each threshold value, an individual neuron predicts the probability whether the observation will be below this value. A sigmoid activation function is selected for each output unit to output a probability and thus a value between 0 and 1. The loss function used here, described by Brier in 1950 [13], is the Brier score

$$BS = \frac{1}{N} \sum_{i=1}^{N} (P_i(T) - O_i(T))^2,$$

which minimizes the averaged quadratic deviation between the binary target values $O_i(T)$ indicating that the observed power $y < T$ and the predicted probabilities $P_i(T)$ given a threshold $T$ for all samples $i = 1, \ldots, N$.

In contrast to the QRNN, the DRNN usually requires a higher number of output units. The range covered by the threshold values of the DRNN often covers more than the individual conditional distributions would need. However, this is necessary in order to be able to make a prediction for every possible conditional distribution.

2.3. Multi-Task Learning Distribution Regression Neural Network mtlDRNN

Typically it is of interest to predict the probabilities of various thresholds in order to discretely approximate the conditional CDF. One rather time consuming way to achieve this is by training an individual model per threshold as in DR. Instead of training an individual neural network per threshold one single neural network can predict all threshold probabilities by using several output units. Since every threshold implies an individual loss function every output unit is concerned with an individual task. Applying various loss functions to one neural network is known as multi-task learning where every task is represented by a distinct loss. Multi-task learning has a regularizing effect, since a subset of the trainable parameters is combined with a larger set of tasks, where every task sets distinct requirements. They are jointly less free and therefore often lead to useful internal representations of the data.

One major drawback of distribution regression is that with many thresholds redundancy is generated, since similar thresholds will generate similar outputs. Since the probabilities are predicted independently it occurs, that a higher threshold has a lower predicted probability than a lower threshold. This is referred to as quantile crossing, which is also a disadvantage of the method and must be corrected by post-processing the model output. With a large number of thresholds, the probability of quantile crossing rises. Opposed to this, a low number of thresholds restrict the resolution with which the predicted distribution is scanned.

The newly proposed approach, multi-task learning distribution regression neural network mtlDRNN, takes up the idea of DRNN. At the same time, it eliminates the disadvantage of the many output units. Instead of giving the neural network more than one output unit, another input unit is added instead. The threshold value for which the prediction is to be made is not determined in advance. Instead, this is communicated to the neural network as an input. Like an ordinary CDF, the neural network can be evaluated for any (threshold) value. This makes it possible, for example, to estimate the distribution function of the power, while the neural network takes into account the dependency on the input variables, such as wind speed and wind direction. To train this model it is necessary to combine as many different threshold values as possible with the samples. In theory, this would result in an extremely large data set that would be very difficult to keep in memory. Instead, only the original record is kept in memory. In each epoch, each sample is combined with a new threshold value that can be drawn from a distribution. To ensure that this lies within the range of the target value, it is advisable to draw it from a uniform distribution within the same value range as the threshold values of the DRNN described before.

Our approach avoids the discrete representation of the CDF: A single output unit can be used to predict the probability of being less than any desired threshold. To achieve this, the desired threshold value is added to the set of input variables. The net is then trained in order to correctly predict the probability of falling below the given threshold. The trained net can therefore be evaluated for any
threshold value and has thus directly approximated the CDF instead of discretely scanning it. The neural network is forced to perform not only a discrete but an infinite number of tasks. This increases the demand for multi-task learning and avoids redundancies in training. As a result, the number of parameters can be lower while reducing the risk of quantile crossing. This approach is referenced hereafter as multi-task learning distribution regression neural networks, short mtlDRNN.

2.4. Further ANN-Setup

A consistent deep neural network architecture across all model approaches was chosen to carry out the experiments. The selected deep neural networks each have an entrance layer with at least three dimensions for wind speed and the wind direction components. With 200, 100 and 50 units, the layers have a decreasing number of neurons. However, the number of units in the output layer depends on the respective approach and which requirements are made on the resolution of the conditional distribution. Deep neural networks often have the problem of vanishing gradients, which describes the effect of the initially low dependence of the loss function on the weights of the first layers and the associated difficulties of optimization.

This leads to a slow convergence and can sometimes lead to worse results than with a shallow net. The Leaky ReLU proposed by Maas [7] was selected for the activation function as it has good properties against this vanishing gradient problem. In addition, the weights were trained with the help of the ADAM optimizer, proposed by Kingma [8], which enables a fast training without having the need of carrying all data in the memory. The standard parameters of the ADAM optimizer were used for all networks. The training was conducted in 500 epochs in which the entire data set was processed 500 times. However, the hyperparameters of the neural network were not optimized, since the aim of the study was not to compare the neural networks, but to combine them with the approaches above. Both the optimizer and the ANNs have been implemented in Matlab and extend the usual range of available tools.

3. Data and Data Pre-Processing

Quantile regression, distribution regression and the proposed CDF multi-task learning approach are compared in a data case study. Operational SCADA-data from a wind turbine from a North Sea offshore wind farm is used. The data comprises 10 minute averages of the wind speed and wind direction measured at the nacelle. Wind speed is taken as a direct input without considering air density influence factors such as temperature and air pressure. To avoid the non-continuous step from less than 360° to above 0° degrees, the wind direction was converted into a direction vector with a sine and cosine component. The wind direction components and wind speed are scaled before use so that the mean value is zero and the standard deviation is one. As an example, the active electrical power was selected as the target value. Each trained neural network will therefore map the relationship between direction and magnitude of the wind vector into a power value and can thus be interpreted as a direction-dependent power curve, which additionally allows a conclusion about the uncertainty of the estimation. The experiment comprises data from the whole year 2016, using the first two quarters and the last quarter as training period, while the remaining quarter is withheld from model training for test purposes.

For this case study the training data was filtered in order to be able to model a failure-free production operation. Samples whose power measurements at wind speeds between 5 and 25 m/s fall below a power of 5% of the nominal power are removed. In addition, all samples in which the turbine was not in "production" mode at wind speeds greater than 5 m/s were removed, for example, to prevent manual operation by maintenance personnel. Furthermore, the adjacent timestamps of the already excluded time periods were removed as a precautionary measure in order to eliminate reduced power samples which appear in the transition from production to shut down or vice versa. For timestamps between 5% and 95% of the nominal power and at wind speeds greater than 5 m/s, curtailment was also detected and removed. For this purpose, two conditions were checked. First, it is checked whether the variation in power over time is less than 30% of the cases that occur. Second, it is tested whether
the wind speed is greater than the 99% quantile of the wind speeds that would occur at the measured power without curtailment. The result of the data cleansing is shown in Figure 2.

When monitoring other performance indicators the filtering has to be adjusted for the specific purpose or alternatively left out altogether which would mean that no prior expert knowledge has to be applied.

![Figure 2. Visualization of the raw and filtered data by plotting the normalized power against the nacelle wind speed measurements.](image)

4. Case-Study Results

Figure 3 shows the results of the models and the measured performance for a selected day in the test period. The output of QRNN, DRNN and mtlDRNN is displayed from the top to the bottom diagram. The blue line shows the measured power, which was normalized with the nominal power of the wind turbine. The orange area represents the predicted conditional distribution, which was converted into a probability density function. For reasons of better visibility, the color values were scaled for each point in time so that the highest probability density corresponds to constant maximum color intensity.

The temporal profile of all models seems to reproduce the temporal profile of the measurement more or less equally well. Between 04:10 a.m. and 01:50 a.m. the turbine reaches nominal power. Here a slight overestimation by QRNN and mtlDRNN and a very sharp prediction by the DRNN are visible. The overestimation by mtlDRNN is accompanied by a greater predicted uncertainty, which is why this is a rather minor problem. From about 02:00 p.m. the wind turbine is experiencing a pitch error. The difference between the predicted distribution and the observed power successfully detects anomalous behavior one hour before. By specifying the conditional density distribution, it is numerically possible to classify this deviation as unusual, while the deviations at the beginning of the day can be classified as in the range of possible. For anomaly detection also the sequence of time stamps with low probability should be taken into account which can be realized by adding a sequential probability ratio test (SPRT).

With regard to the density function, there are differences between the three models at different points in time. It can be seen that the vertical color gradient is not always continuous. This applies to the DRNN and especially to the QRNN, while the color gradient of the mtlDRNN makes a continuous appearance. In the case of the QRNN, the non-continuous behavior runs parallel to the power measurement, but in the case of the DRNN it seems to run parallel to the x-axis. The explanation can...
be found in the weaknesses of the output units. The individual outputs of the QRNN are the quantiles, which ideally run parallel to the performance. The output unit of the DRNN refers to one threshold value each. If individual output units are underestimated or overestimated, the strips described in the display appear. The vertical red dashed line marks August 7, 2016 at 01:00 am. For this point in time, the CDF and the corresponding probability density of normalized power are shown in Figure 4.

**Figure 3.** Progress of the power (normalized with the nominal power) and the predicted conditional density distribution.
While there are hardly any differences between the CDFs of the individual models, there are clear
differences in the probability density function. This is even more surprising as the CDF is the integral
part of the PDF. A closer look at the CDF reveals that in the case of the QRNN and the DRNN a slight
jagged pattern is visible. This results in short successive changes in the gradient and thus leads to the
large variations in the PDF. In the case of the mtlDRNN it comes to a clearly smoother development.
This behavior is much more realistic, as it can be expected that a power measurement should have
approximately the same probability of occurrence as a minimally larger or smaller power
measurement. If anomalies on the basis of probability density are to be detected, in the worst case it
may lead to a power measurement being marked as anomaly due to a low probability density, while a
directly adjacent power value with a greater density is classified as normal.

![CDF and PDF](image1.png)

**Figure 4:** Cumulative distribution and probability density functions derived by the QRNN, DRNN
and mtlDRNN for 01:00 a.m., 7th August 2016.

Probabilistic models of this type are usually evaluated using the continuous ranked probability
score (CRPS) as suggested by Brown in 1974 [14]. This is a global score that compares the entire CDF
with the sample and thus evaluates the distribution not only at the sample location, as a local score
would. The CRPS calculates by modelling the CDF of the measurement as a Heaviside function. The
difference between the predicted CDF and the measured Heaviside function is squared and the integral
from minus to plus infinity is then evaluated as a measure for the sign-independent error area.

This process is repeated for each sample and the determined integrals are averaged. Figure 5 shows
a boxplot with the CRPS values of the different models. The individual boxes contain 100 block
bootstrap estimates of the test set CRPS with a block length of 7 days. The test period is divided into 7
days long sections, which are numbered consecutively. For each bootstrap CRPS, as many blocks are
selected randomly from these blocks as are available in the test set. Then the CRPS is determined for
each bootstrap set and this process is repeated 100 times. The distribution of the bootstrap CRPS
indicates the uncertainty of the CRPS estimate. Since the distributions cover almost the same range, it
can be concluded that there is no significant difference between the models in the CRPS.
A local score is the ignorance score as described by Roulston in 2002 [15]. It has a close relationship to cross entropy or the logic-likelihood function and is based on the principles of information theory. The calculation is based on the predicted probability of observing the corresponding measurement. The negative logarithm of this local probability is then averaged over all samples. To calculate the probability, the range of power values was discretized into 200 equidistant sections and the probability density was integrated over each sub-interval. The ignorance score is significantly stricter for samples with predicted probabilities close to zero, as they approach infinity in the negative logarithm. Since the predicted probabilities of the models repeatedly came close to zero, these probabilities were set to a thousandth of the smallest predicted probability of another model.

Figure 6 shows the distribution of the 100 block bootstraps of the ignorance score of the respective models. The use of mtlDRNNs results in significantly fewer errors than in the case of DRNN. DRNN, on the other hand, clearly has smaller errors than the QRNN.

5. Conclusions
The methods investigated allow probabilistic models, based on deep neural networks, to be used for the detection of anomalies in wind turbines. We propose a new method for a continuous approximation of conditional distribution functions. This approach avoids typical weaknesses of models such as QR or DR: On the one hand, inconsistencies such as quantile crossing are avoided. On the other hand, high-dimensional and extensive data sets can be processed using mini-batch and artificial neural networks. This is made possible by avoiding redundant parameters in the model through the use of multi-task learning. The approach allows information to be exchanged between individual quantiles in order to simultaneously train a leaner model. It may be applied to monitor different kinds of performance indicators for predictive maintenance and is not limited to a specific time resolution. The approach advances the probabilistic output of the anomaly detection system by providing a more realistic probability density function and avoiding misdetections due to local effects. The model has been tested and approved by the use of real-world data from an offshore wind turbine.

In the example shown, information on wind speed and direction was used in the neural network. However, the application is not limited to these variables. Depending on the extent of the training data, many more variables can also be taken into account. Quantile and distribution regression in neural network applications are limited in resolution. These procedures sometimes have a partly questionable, volatile profile in respect of probability density. In addition, an increased number of output units is required in the ANN. Predicting multidimensional target values, would mean that the number of needed output units is proportional to the number of estimated quantiles or threshold values times the
dimension of the target value. The approach presented here, on the other hand, could represent each additional target variable dimension with an additional output unit and an additional input unit, thus greatly limiting the complexity.

Acknowledgement

The work that has led to this publication was carried out within the research project ModernWindABS (FKZ 0324128), which is being funded by the German Federal Ministry for Economic Affairs and Energy.

References

[1] Janssens, Noppe, Devriendt, Walle, & Hoecke. (2016). Data-driven multivariate power curve modeling of offshore wind turbines. Engineering Applications of Artificial Intelligence, 55, 331-338.
[2] Carmona, M., & Sanz-Bobi, M. (2016). Normal power generation area of wind turbines for the detection of abnormal performance. Renewable Energy Research and Applications (ICRERA), 2016 IEEE International Conference on, 335-340.
[3] Sun, Li, Wang, & Lei. (2016). A generalized model for wind turbine anomaly identification based on SCADA data. Applied Energy, 168, 550-567.
[4] Zhou, Y., Hu, W., Min, Y., Zheng, L., Liu, B., Yu, R., & Dong, Y. (2017). A semi-supervised anomaly detection method for wind farm power data preprocessing. Power & Energy Society General Meeting, 2017 IEEE, 2018, 1-5.
[5] Zhao, Liu, Hu, & Yan. (2018). Anomaly detection and fault analysis of wind turbine components based on deep learning network. Renewable Energy, 127, 825-834.
[6] Bangalore, & Patriksson. (2018). Analysis of SCADA data for early fault detection, with application to the maintenance management of wind turbines. Renewable Energy, 115, 521-532.
[7] Maas, Andrew L., Awni Y. Hannun, and Andrew Y. Ng. "Rectifier nonlinearities A reference improve neural network acoustic models." Proc. icml. Vol. 30. No. 1. 2013.
[8] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization.", 3rd International Conference for Learning Representations , San Diego, 2015,arXiv preprint arXiv:1412.6980 (2014).
[9] Taylor, James W. "A quantile regression neural network approach to estimating the conditional density of multiperiod returns." Journal of Forecasting 19.4 (2000): 299-311.
[10] Koenker, Roger, and Gilbert Bassett Jr. "Regression quantiles." Econometrica: journal of the Econometric Society (1978): 33-50.
[11] Cannon, Alex J. "Quantile regression neural networks: Implementation in R and application to precipitation downscaling." Computers & geosciences 37.9 (2011): 1277-1284.
[12] Peracchi, Franco. "On estimating conditional quantiles and distribution functions." Computational statistics & data analysis 38.4 (2002): 433-447.
[13] Brier, Glenn W. "Verification of forecasts expressed in terms of probability." Monthey Weather Review 78.1 (1950): 1-3.
[14] Brown, Thomas A. "Admissible scoring systems for continuous distributions." Manuscript P-5235, The Rand Corporation, Santa Monica, CA, 22 pp. [Available from The Rand Corporation, 1700 Main St., Santa Monica, CA 90407-2138.]
[15] Roulston, Mark S., and Leonard A. Smith. "Evaluating probabilistic forecasts using information theory." Monthly Weather Review 130.6 (2002): 1653-1660.