8th International Conference
on
Wind Turbine Noise
Lisbon – 12th to 14th June 2019

Is it possible to predict background noise levels from measured meteorological data with machine learning techniques?

Alexis Bigot, Sixense Environment: alexis.bigot@sixense-group.com
Guillaume Hochard, Quantmetry: ghochard@quantmetry.com

Abstract

Background noise levels play a major role when it comes to ensuring compliance with French noise regulation. Background noise levels are usually assessed based on scatter plots: noise levels vs wind speeds and/or wind directions.

In this article, we propose to model the background noise levels using machine learning techniques.

The datasets are built with 10 minutes meteorological data as well as background noise level measured in dBA.

We used gradient boosting which is a supervised machine learning technique used in classification and regression problems. The objective is to train the model for each dataset, and evaluate the accuracy of the regression algorithm.

The results are very promising: the mean absolute error (MAE) of the prediction are 1.07 dBA on dataset A and 1.71 dBA on dataset B. We are convinced that this technique will change the way we manage noise and meteorological data in acoustics and wind energy.

1. Introduction

Background noise levels play a major role when it comes to ensuring compliance with French noise regulation. Background noise levels are assessed using scatter plots: noise levels vs wind speeds. This statistical analysis is carried out for several “homogeneous conditions” as defined in the French normative [1]:

- Day/night
- Wind direction sectors
- Time of the day
- Human activities
- Meteorological conditions
- Seasons

The problem is that this methodology is not precise enough if we need an accurate model of the background noise vs meteorological conditions: the standard deviation is always greater than 3 dBA. First, we should take into account all the meteorological conditions available (not only wind speed and the wind direction as shown in figure 1). Secondly we need new techniques, with a better accuracy than the classical scatter plots.

That’s why we decided to test machine learning techniques for a better assessment of noise levels around wind farms.

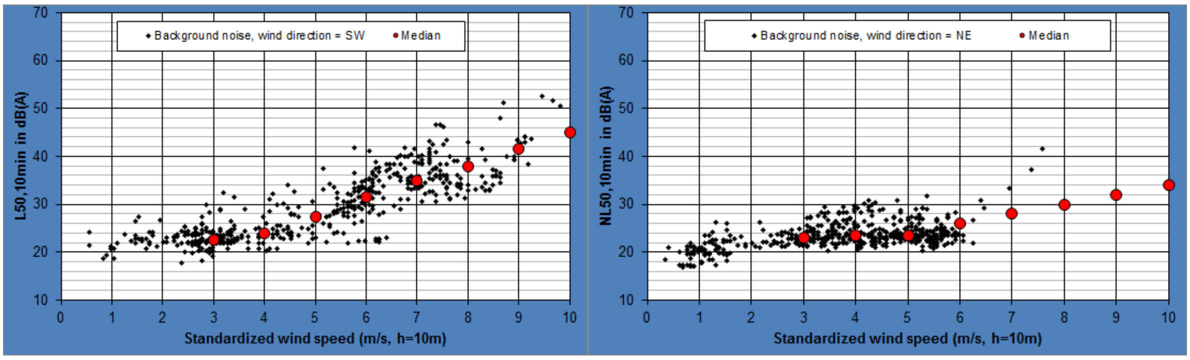

Figure 1: Example of scatter plots for night period (22h-7h), for two homogeneous classes: Wind direction = South-West [135° - 315°[and wind direction = North-East [315° - 135°[. In red = calculation of the median or extrapolation of the trend of background noise.

2. Dataset & methodology

2.1 Dataset

During previous WTN conferences we showed that the wind speed gradient and the temperature difference had an influence on the background noise levels [2]. Those parameters also drive the refraction during noise propagation of wind turbine noise [3].

Based on this observation we decided to choose two datasets that include wind speed and temperature measured for at least two different heights.

In the dataset A, the input features are:

- Datetime, from 11/23/2017 to 12/14/2017
- Temperature measured at a height of 1.5 m
- Temperature measured at a height of 10m
- Wind speed measured at a height of 10m
- Wind speed measured at a height of 100m (hub height of the future turbines)

- Wind direction
- Relative humidity
- Point of acquisition

and the measured noise level, the target to model.

In the dataset B, the input features are:

- Datetime, from 05/03/2017 to 06/02/2017
- Temperature measured at a height of 24.9m
- Temperature measured at a height of 99.5m
- Wind speed measured at a height of 24.9m
- Wind speed measured at a height of 99.5m (hub height of the future turbines)
- Wind direction
- Relative humidity
- Point of acquisition

and the measured noise level, the target to model.

In both cases the output data are background noise level L50,10min in dBA, measured at several locations around the wind farm project: there are 9 location of noise measurements in dataset A and 5 in dataset B.
We decided to use only the night time period (22h-7h) of the dataset, because French regulation is more restrictive during night time.

Datetime	Temperature at 1.5m (°C)	Temperature at 10m (°C)	Wind speed measured at 10m (m/s)	Wind speed measured at 100m (m/s)	Wind direction (°)	Relative humidity (%)	Measure point	Noise level
11/23/2017 00:00	12.2	12.4	9.0	14.2	164	67.9	1	37.7
11/23/2017 00:10	12.4	12.5	9.7	13.6	163	67.4	1	36.6
11/23/2017 00:20	12.5	12.5	9.4	13.6	164	67.1	1	36.4
11/23/2017 00:30	12.6	12.7	9.7	13.6	167	66.1	1	32.5
11/23/2017 00:40	12.7	12.8	10.2	14.2	167	66.1	1	31.4
11/23/2017 00:50	12.7	12.8	10.1	15.0	167	65.9	1	32.8

Figure 2: First rows of dataset A

2.2 **Methodology**

To model this dataset, we propose to use a supervised machine learning approach.

Supervised machine learning is part of artificial intelligence techniques which aims at learning from a set of features a decision pattern to predict target values. A supervised machine learning model thus implies two phases, the learning and the inference phase.

In the learning phase, a model is trained on a set of samples that includes prepared features and corresponding targets in order to make the most accurate predictions. Once the model is trained, we can use it to infer new target values based on a dataset of new samples with the same set of features used for the training phase.

Gradient boosting is a supervised machine learning technique used in classification and regression problems based on the concept of ensembling, i.e. combining weak learners to produce a prediction model. The model is built sequentially, first producing a first model which performance will be evaluated.
The prediction errors are then weighted in order for the next model to correctly predict the difficult samples that were incorrectly predicted by the previous model. This process is repeated iteratively for a given number of rounds.

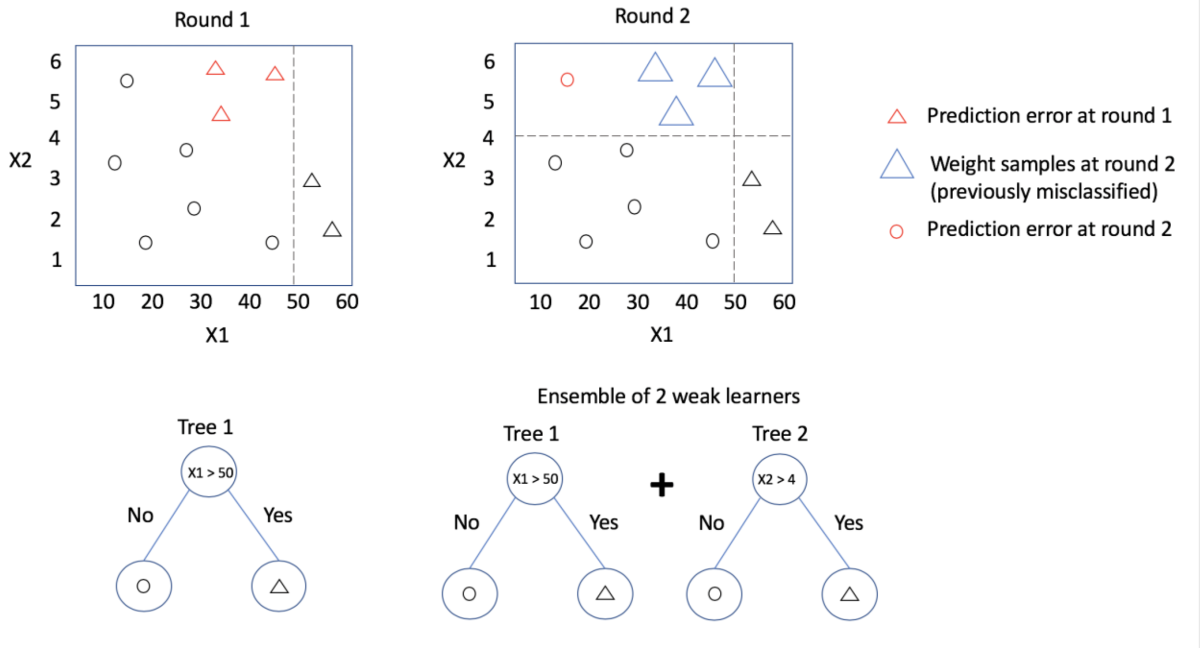

Fig. 3 Gradient boosting explained simply for a supervised classification example. Decision trees are iteratively built in order to make better predictions for the incorrectly predicted values by the previous tree.

For this paper, we have used XGBoost, a popular and efficient open-source implementation of gradient boosting [4]. XGboost is famous for winning machine learning competitions and because it adapts to a large variety of data types and offers a broad palette of hyperparameters that can be tuned to enhance model performance. This kind of model is thus a reliable choice for supervised machine learning regression problems, like the one presented in this paper.

One problem that could occur during the training phase is overfitting. This term means that the model complexity is too high and has learned “by heart” the training set. The effect of overfitting is straightforward on new data, as the performance measures on new predictions get lower than those on the training set. To overcome this problem, we split the data in two sets - a training and a validation set - with a 80/20% ratio. To assess model generalization and performance on new data, we will compare the predictions errors on these two sets.

One other way to control overfitting is by performing cross-validation on the training set. This process involves to randomly partition the training data into several folds. One of the folds will be left out the training set and will be used for performance evaluation. This process is repeated for every fold and validation results are averaged over the different validation rounds to get a better evaluation of the model's predictive capability.

To evaluate the performance, we use the mean absolute error and the standard deviation error. Given a predicted value $\hat{y}_i$ for and the corresponding ground truth value y_i, the mean absolute error over the n samples in the dataset is defined as:

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

3. Results

3.1 Global results

The machine learning models trained on these two datasets offer good performances, as illustrated in Table 1, with low bias (1.10 dBA and 1.82 dBA on the test sets) and limited variance between the training and the validation sets (resp. 0.79 dBA and 1.16 dBA on dataset A and B), which indicates good generalization of the models on data unseen during the training phase, and thus limited overfitting.

	Train set		Test set		Test/train comparison
Dataset	Mean Abs Error (1)	Std Dev.Error	Mean Abs Error (2)	Std Dev. Error	Error difference (2) – (1)
A	**0.29 dBA**	0.38 dBA	**1.07 dBA**	1.45 dBA	**0.78 dBA**
B	**0.60 dBA**	0.83 dBA	**1.71 dBA**	2.42 dBA	**1.11 dBA**

Table 1. Performance of the two models on the training and test sets

Figures 4 and resp. 5 present the predicted values over the ground truth values for all the samples in the training and test sets. The difficult values to predict in the test set are clearly identified as outliers from the linear fit. Further exploration on the prediction errors could allow to better understand in which conditions the model is less accurate. We can also compare the generalization power of the two models by observing the distribution of errors for the train and test sets: more variance is observed on the training dataset A than on the training dataset B, and we observe the same pattern in the corresponding test sets.

On dataset B, we observe greater prediction errors for larger values. This could be explained by the fact that the model has less data to be trained on for large values, as we clearly observe lower density for large values on the test set evaluation scatter plot in Figure 5.

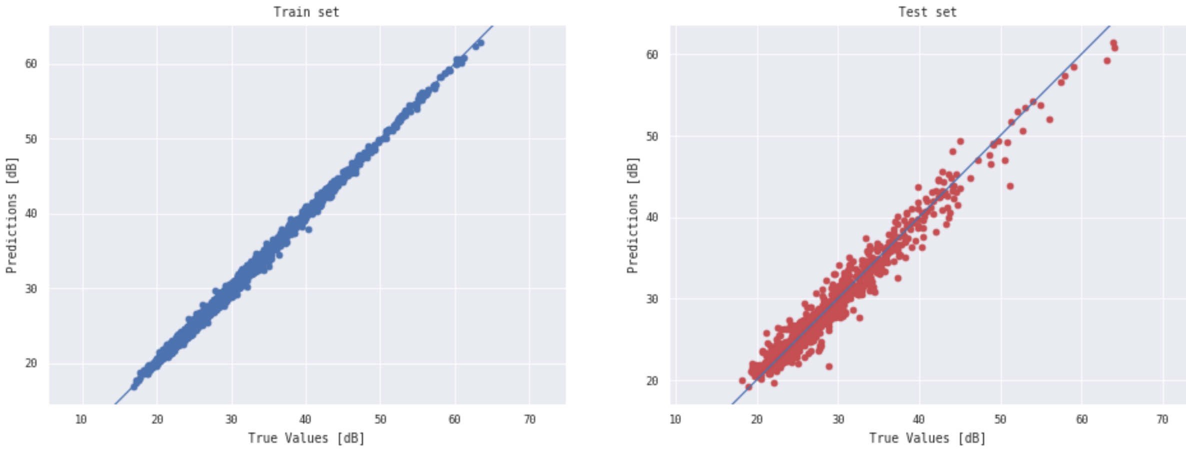

Figure 4: Predictions and ground truth values for training and validation set on dataset A

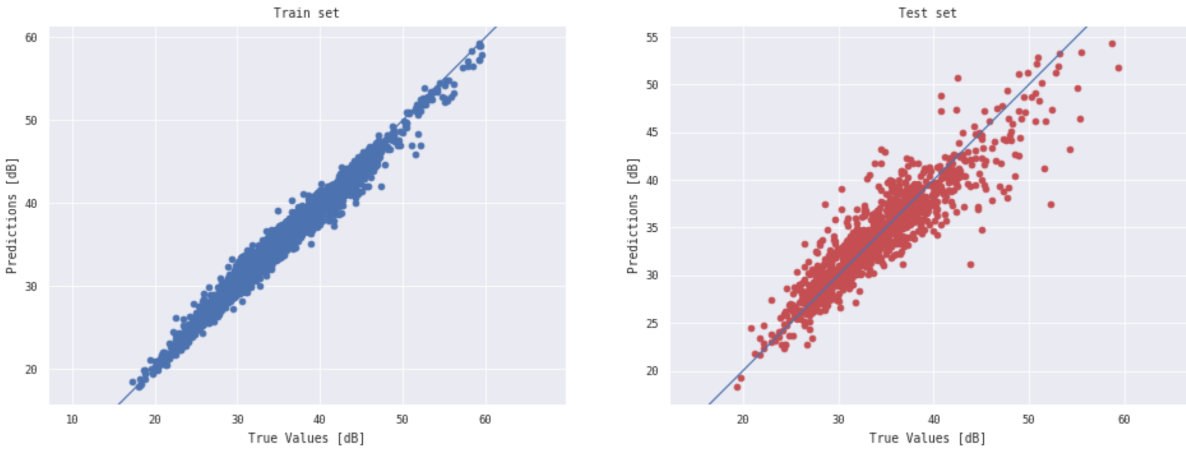

Figure 5: Predictions and ground truth values for training and validation set on dataset B

3.2 **Feature importance**

One advantage of using gradient boosting trees methods in the explainability of the decision making process performed by the model, both at a global scale (understanding which feature is important in the model) and at a local scale (which feature contributed quantitatively to drive the target value prediction in a specific direction). Local scale interpretability can be obtained using methods as shapley values [5].

At the global scale, the feature importance is an insightful tool to assess the predictive power of features. In the context of gradient boosting trees, it measures and computes the average reduction in impurity across all trees in the ensemble of weak learners due to each feature. Therefore, features that are used early in the tree construction (closer to the root node) get larger importance value.

By plotting the feature importance of all features in Figure 6 and 7 for the two models, we observe that the top most important features in the dataset A are the temperature at 1.5m, the mean wind speed at 10m/100m and the relative humidity. In the dataset B the important meteorological features are similar. The point of acquisition (area to predict the noise level) is also of prime importance, which confirms the fact that the noise level depends on the meteorological conditions but also the location of the measure itself.

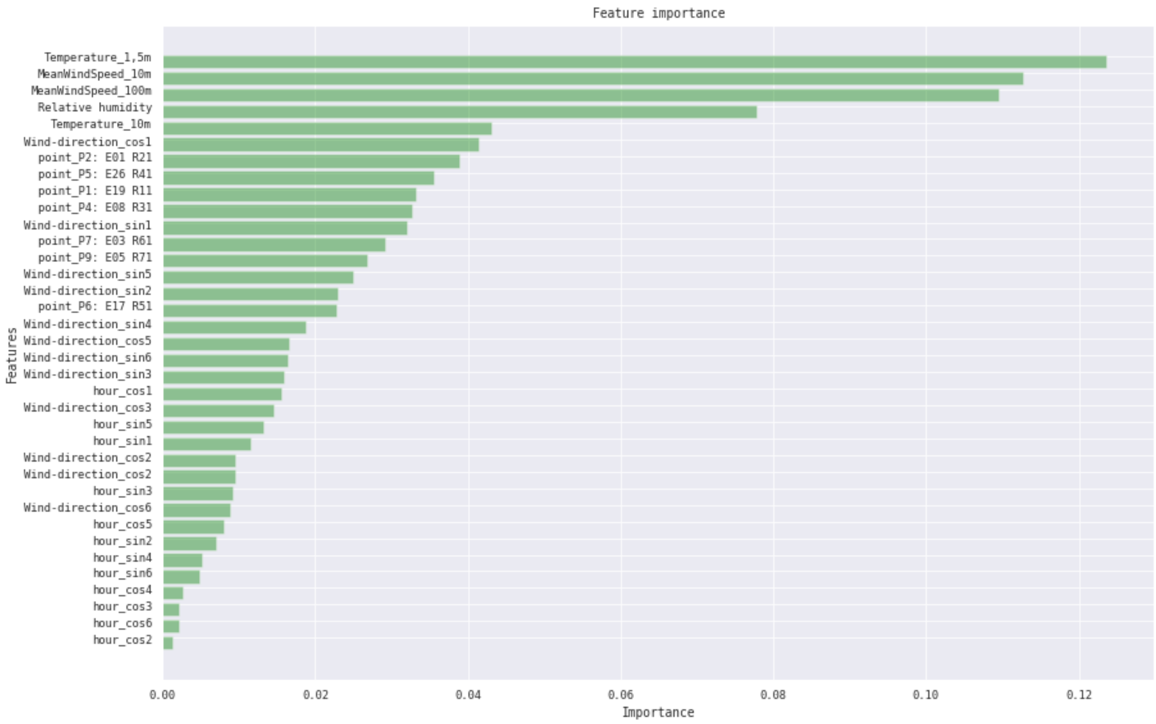

Figure 6: Features importance on dataset A

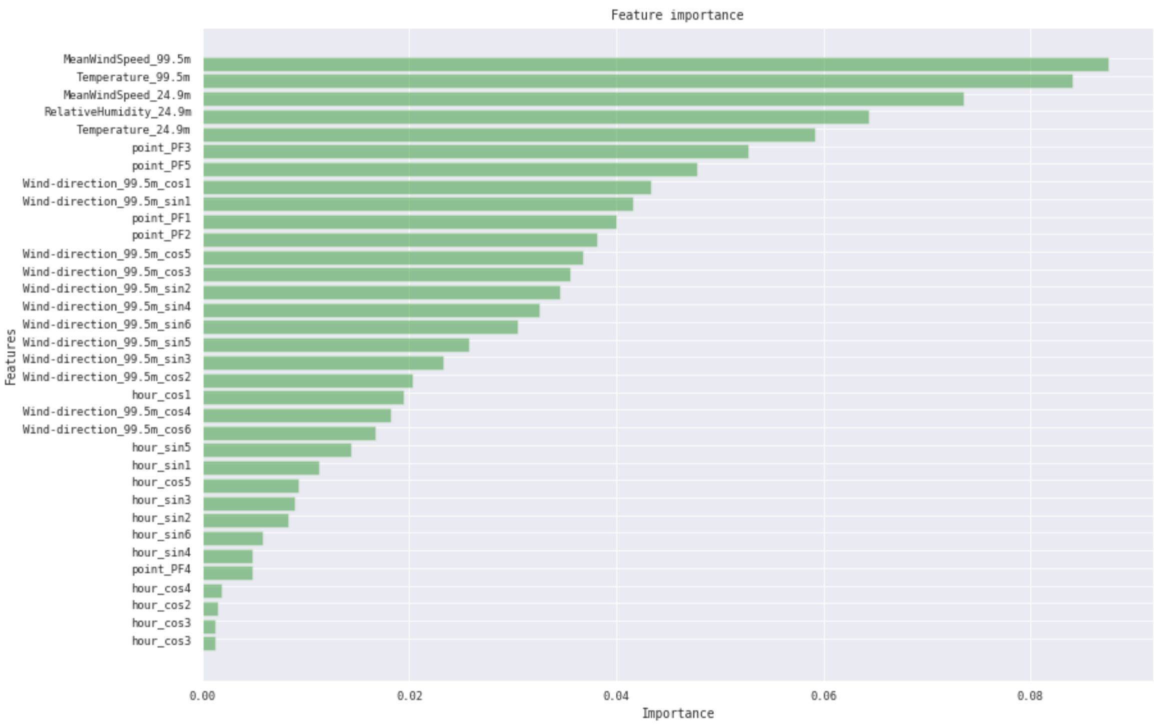

Figure 7: Features importance on dataset B

In parallel of this machine learning approach, we tried to benchmark these results to neural networks based models, but these models gave slightly less performance than the one obtained with the gradient boosting trees models. Nevertheless, they may give good results and should be tested again if the dataset is very large.

4. Discussion

Machine learning techniques give good results on wind turbine noise prediction, but further investigation are necessary in order to draw more reliable conclusions. The questions are: is it possible to extrapolate from one site to another? Is it possible to model and predict the trend of background noise over one year? and what are the minimum parameters and amount of for that scope?

In the near future we can imagine several applications of machine learning in wind turbine noise. Some use cases could be:

- Improving the modelization and understanding of background noise during impact study.
- Extrapolation of missing data at one noise location, in base of meteorological data and noise measurements at other locations around the site.
- Estimation of background noise during the operation of wind turbines, when they can't be stopped. A machine learning algorithm could be implemented in the operating system in order to evaluate in real-time the noise emergences in the neighborhood.
- Optimization of the energy production with respect to meteorological predictions including the noise criteria.

References

[1] Projet de norme AFNOR NF S 31-114 - Version du 7/7/2011 – v3

[2] Bigot, A.; Slaviero, D.; Mirabel, C.; Dutilleux, P. (2015). *Influence of vertical temperature gradient on background noise and on long-range noise propagation from wind turbines*. 6th International Conference on Wind Turbine Noise.

[3] Bigot, A.; Economou, P.; Economou, C. (2017). *Wind turbine noise prediction using Olive Tree Lab Terrain*. 7th International Conference on Wind Turbine Noise.

[4] CHEN, Tianqi et GUESTRIN, Carlos. *Xgboost: A scalable tree boosting system*. In : Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. ACM, 2016. p. 785-794.

[5] LUNDBERG, Scott M. et LEE, Su-In. A unified approach to interpreting model predictions. In : Advances in Neural Information Processing Systems. 2017. p. 4765-4774.