Short Term France Wind Production Forecasting Based on Meteorological Open Source Data and Deep Learning

Hugo Antoine François HAMBURGER – s212364

Ole Winther - DTU Compute, Technical University of Denmark

Introduction :

The idea of the project is to use of a huge quantity of non-professional French weather data which allows a strong granularity (as there is a lot of dispersed stations) instead of being restricted to a few weather stations from the national weather agency (~50) to forecast national Wind Energy production on a short term (H+1, H+2, H+3, H+4).

The data consist of temperature, sky visibility, humidity, wind chill, maximum wind speed, pressure and location with an hourly time step (provided by ~800 stations dispersed unequally over the French territory). Each of these stations can have missing data and didn't measure necessarily all the mentioned parameters.

Input Data Building :

A first step was to limit the number of data used in order to keep a reasonable computing burden. Here I chose to only consider wind speed to limit the computing burden and as most of the wind production forecasting projects use only wind speed data. If an expansion is possible, the other interesting data would probably be maximum wind speed, pressure and temperature as they all impact the wind production. An other restriction was to use approximately 8 month of data in order to cover the seasonality.

The second step concern the handling of the missing values. Here I choose to not consider the stations with to much missing data and to estimate the remaining missing data by using sklearn IterativeImputer function. It results in only 178 stations.

Finally the model is fed with data corresponding of 5086 samples/hourly time steps and each input contain the wind speed measurement between t and t-23h resulting in 24*178 = 4272 features per sample. The output of the model correspond to the wind power production of the next four hours (H+1, H+2, H+3, H+4) resulting in 5086 * 4 points.

Structures of the model evaluated and Results :

I considered 3 neural network structure types :

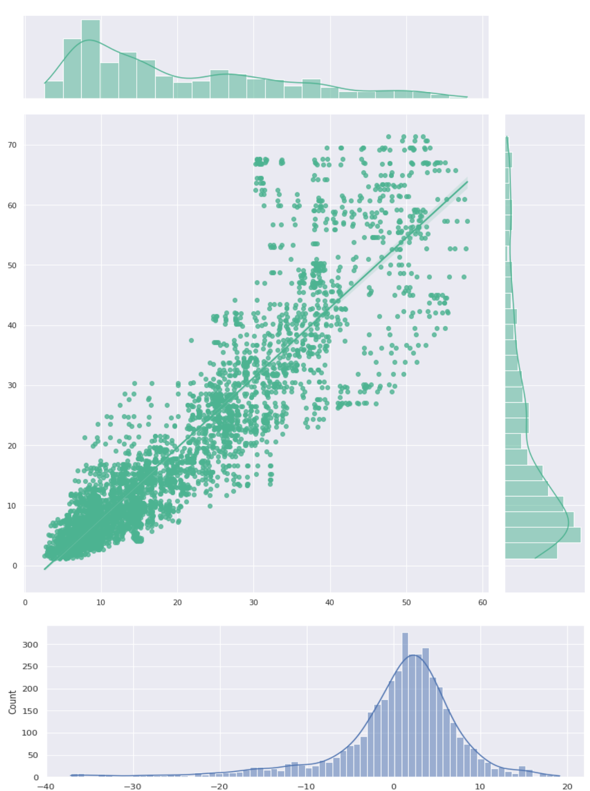
1. A combination of **Dense Layer** and **Regularization Layer** (mainly Dropout)
2. A combination of a first stage of **Convolutional Layers** followed by **Dense Layer and Regularization Layer** (mainly Dropout)
3. A combination of a first stage of **Convolutional Layers** followed by an **LSTM Layer** then by **Dense Layer and Regularization Layer** (mainly Dropout)

The idea behind the use of **Convolutional Layer** and **LSTM Layer** was to try to improve the comprehension of the time dependency between successive values and this types of layers are widely used for time series forecasting.

The models that obtain the best results for each structure types are presented below. Each model type structure were tested for different layers organization and hyperparameters. It is important to notice that the second and the third types represent many more parameters and where trained under more constraining parameters (batch size, kernel size and epoch number) to keep a reasonable time (from ~20 min to ~60 min) and memory usage. Sometimes the memory limit was also an issue that enforce stronger arbitrages.

Layer (type)	Output Shape	Param #
dense_85 (Dense)	(None, 1024)	4375552
dropout_72 (Dropout)	(None, 1024)	0
dense_86 (Dense)	(None, 512)	524800
dropout_73 (Dropout)	(None, 512)	0
dense_87 (Dense)	(None, 256)	131328
dropout_74 (Dropout)	(None, 256)	0
dense_88 (Dense)	(None, 128)	32896
dropout_75 (Dropout)	(None, 128)	0
dense_89 (Dense)	(None, 64)	8256
dropout_76 (Dropout)	(None, 64)	0
dense_90 (Dense)	(None, 32)	2080
dropout_77 (Dropout)	(None, 32)	0
dense_91 (Dense)	(None, 16)	528
dropout_78 (Dropout)	(None, 16)	0
dense_92 (Dense)	(None, 8)	136
dropout_79 (Dropout)	(None, 8)	0
dense_93 (Dense)	(None, 4)	36

Total params: 5,075,612
Trainable params: 5,075,612
Non-trainable params: 0



Bias distribution

Best type 1

Layer (type)	Output Shape	Param #
conv1d_6 (Conv1D)	(None, 4267, 32)	224
max_pooling1d_6 (MaxPooling1D)	(None, 2133, 32)	0
conv1d_7 (Conv1D)	(None, 2128, 32)	6176
max_pooling1d_7 (MaxPooling1D)	(None, 1064, 32)	0
flatten_3 (Flatten)	(None, 34048)	0
dense_20 (Dense)	(None, 512)	17433088
dropout_17 (Dropout)	(None, 512)	0
dense_21 (Dense)	(None, 256)	131328
dropout_18 (Dropout)	(None, 256)	0
dense_22 (Dense)	(None, 128)	32896
dropout_19 (Dropout)	(None, 128)	0
dense_23 (Dense)	(None, 64)	8256
dropout_20 (Dropout)	(None, 64)	0
dense_24 (Dense)	(None, 32)	2080
dropout_21 (Dropout)	(None, 32)	0
dense_25 (Dense)	(None, 16)	528
dropout_22 (Dropout)	(None, 16)	0
dense_26 (Dense)	(None, 8)	136
dropout_23 (Dropout)	(None, 8)	0
dense_27 (Dense)	(None, 4)	36

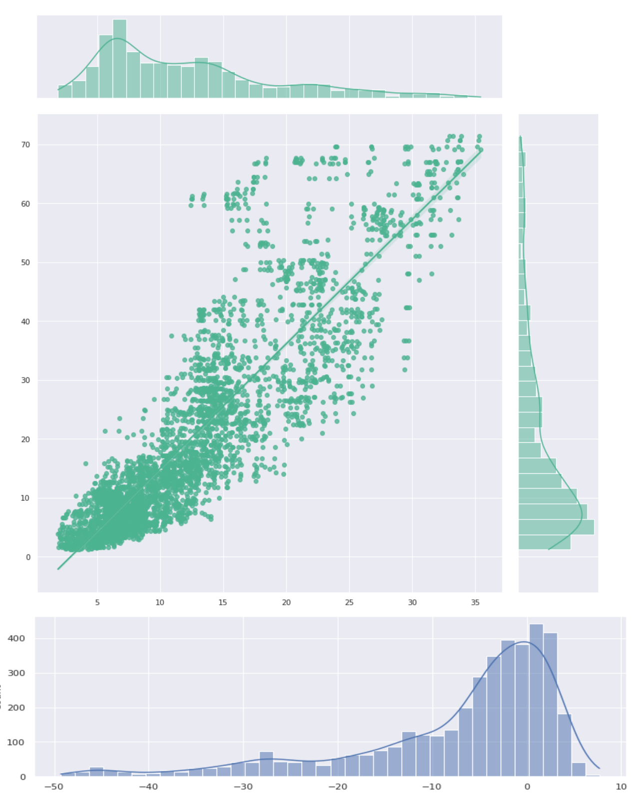
Total params: 17,614,748
Trainable params: 17,614,748
Non-trainable params: 0

Bias distribution

Best type 2

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 4267, 32)	224
max_pooling1d (MaxPooling1D)	(None, 2134, 32)	0
conv1d_1 (Conv1D)	(None, 2129, 32)	6176
max_pooling1d_1 (MaxPooling1D)	(None, 1065, 32)	0
lstm (LSTM)	(None, 1065, 64)	24832
flatten (Flatten)	(None, 68160)	0
dense (Dense)	(None, 128)	8724608
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 64)	8256
dropout_1 (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 32)	2080
dropout_2 (Dropout)	(None, 32)	0
dense_3 (Dense)	(None, 16)	528
dropout_3 (Dropout)	(None, 16)	0
dense_4 (Dense)	(None, 8)	136
dropout_4 (Dropout)	(None, 8)	0
dense_5 (Dense)	(None, 4)	36

Total params: 8,766,876
Trainable params: 8,766,876
Non-trainable params: 0



Bias distribution

Best type 3

	R2	Mean Bias	RMSE	MAPE
Type 1	0.799	0.397	56.684	44.082
Type 2	0.551	-4.980	126.593	38.217
Type 3	0.364	-7.369	179.452	40.840

Result analysis and Dicussion

Model including LSTM and Convolutional layers did not perform as well as expected compared to what can be found in the literature but it can be explained by the way stronger restrictions on the hyperparameters. An other possibility is that they are less suited for this dataset. Nevertheless, convolutional model seems the most promising among those two especially when considering the fact that the considered batch number was low and could be increase without generating to big running time. It could be interesting to try to reduce the stations and consider only the most relevant one to allow to release a bit the hyperparameters.

Finally the best models found for this exercise were composed only of Dense Layers and Regularization Layers. We obtain an acceptable Bias and a R2 coefficient of 0.8. it is still probably not good enough to use it in practical case. It seems difficult to me to improve the model just by changing hyperparameters and the number of layers but including other data type can be promising, I'm especially thinking at temperature and pressure.

To sum up, the main contribution of this project is to partially explore a Dataset that has never been used for machine learning and that consist of a higher number of lower quality data compared to what is usually used. Several conventional model structure for wind forecasting were tested and structure type 1 is the one that perform the best.

The potential extensions of this work would be to use other types of weather data types related to the wind production. In order to keep a reasonable amount of data. It could be interesting to consider only the stations that contribute the most to the model, this make sense as the wind production in France is mainly located in the northern part of country. Therefore, there is probably a part of the stations that is irrelevant. It is also possible to try to forecast the demand of the country in the same way using mainly temperature (electricity demand in France is highly thermosensitive) .

Bibliography

[1] Hong, Y.-Y., & Rioflorido, C. L. P. P. (2019). A hybrid deep learning-based neural network for 24-h ahead wind power forecasting. In Applied Energy (Vol. 250, pp. 530–539). Elsevier BV.

[2] Wu, Z., Luo, G., Yang, Z., Guo, Y., Li, K., & Xue, Y. (2022). A comprehensive review on deep learning approaches in wind forecasting applications. In CAAI Transactions on Intelligence Technology (Vol. 7, Issue 2, pp. 129–143). Institution of Engineering and Technology (IET).

[3] Li, P., Wang, X., & Yang, J. (2020). Short-term wind power forecasting based on two-stage attention mechanism. In IET Renewable Power Generation (Vol. 14, Issue 2, pp. 297–304). Institution of Engineering and Technology (IET).

[4] Deng, X., Shao, H., Hu, C., Jiang, D., & Jiang, Y. (2020). Wind Power Forecasting Methods Based on Deep Learning: A Survey. In Computer Modeling in Engineering & Sciences (Vol. 122, Issue 1, pp. 273–301). Computers, Materials and Continua (Tech Science Press).