Internship report

Contents

Abstract

For day-ahead forecasts, the combination of Numerical Weather Prediction (NWP) models and post-processing algorithms is the most effective method.
However, it is hard to extract from all the literature on the subject the best algorithm to use because of the lack of consistency in the different approaches.

During my Internship, my mission was to investigate the best algorithms according to the literature so as to improve the day-ahead irradiance forecasts.
My final results demonstrated improved metrics in comparison to the current algorithm used by Reuniwatt.

1 Introduction

1.1 Background and motivation of the intership project

This report is the result of my 6-month internship that took place in Reuniwatt, a leader in cloud observation and forecasting. My internship extended from March 1st to August 31st, taking place during the second semester of my academic gap. The main subject of the intership was the post-processing of the day ahead NWP irradiance forecasts. Despite their proven utility for day-ahead irradiance forecasting, NWP models predictions can still be improved thanks to post-processing. As i will show in **??**, many models have been investigated in the literature, and it's thus important to draw a clear benchmark of all the available state-of-the-art models.

1.2 Objectives of the internship

Hereafter the main objectives of the internship:

- Benchmark several models on the post-processing of a single NWP model.
- Sensitivity study of the models.
- Comparison of the results with the current model used by Reuniwatt for day-ahead forecasting.

2 Methodology

2.1 Data source

Verbois et al. demonstrated that using a large set of predictors can significantly improve the performances of post-processing models, while Suksamosorn et al. selected WRF forecasts of irradiance, temperature, relative humidity and the solar zenith angleas relevant inputs of the models.
The forecasted data is for each day the one relative to the origin 00:00 UTC of the day before. Our initial data source for the forecasts was GFS, and we opted for the following set of predictors, easily available for any location:

ghi_{GFS}	T^{2m}_{GFS}	θ	ϕ	ghi_{cs}
Irradiance forecasted	Temperature forecasted 2 meters above the ground	Zenith angle	Azimuth angle	Clear-sky irradiance

Table 1: Set of predictors.

Verbois et al. advises researchers to analyze theirs models' performances over several years but I was at this point limited by the Reuniwatt API, thus I opted initially for learning during 2020 and testing during 2021.
The four initial study sites are the following:

2.2 Metrics

Even if papers like Mayer and Yang state that the correlation coefficient is the recommended metrics to use when no clear directive is given, the metrics that I am going to investigate are the one already prefered by Reuniwatt, The mean absolute error (MAE) and the root mean square error (RMSE).

- The mean absolute error

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|I_{forecast,i} - I_{measure,i}|$$

- The mean bias error

$$MBE = \frac{1}{N}\sum_{i=1}^{N}(I_{forecast,i} - I_{measure,i})$$

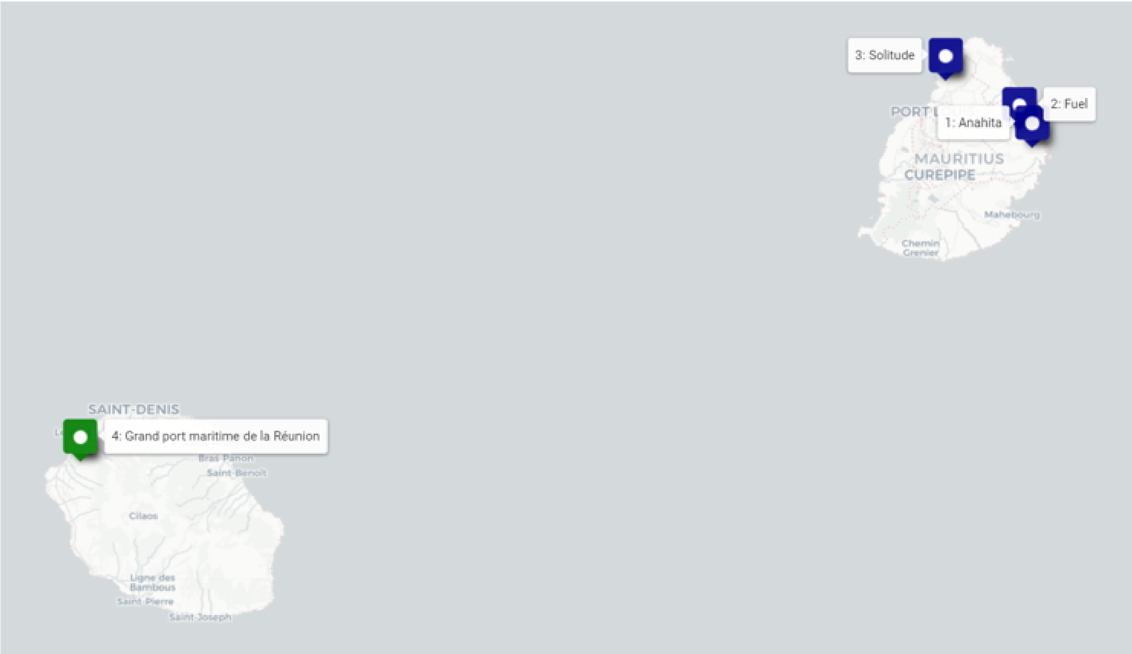

Figure 1: Four initial study sites

- The root mean square error

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(I_{forecast,i} - I_{measure,i})^2}$$

- The skill score s of a certain accuracy measure A, with R denoting the reference irradiance

$$s = 1 - \frac{A(X,Y)}{A(R,Y)}$$

I also wanted to investigate an MBE optimization, but the results were not convincing and MBE is more seen in our study as a metrics to verify after post-processing. We indeed aim at the lowest absolute MBE.

2.3 Models investigated

Our bibliography study leads us toward the most relevant models to be tested. Concerning the reference model, Lorenz et al. and others suggested the use of the persistence model, consisting in taking as prediction the latest measure available,

but this model turned out to have too poor results to be a good reference model. I opted for using the raw forecasted value as the reference model.
Suksamosorn et al. proposed a really interesting linear model based on a Kalman filter scheme. Hence the Kalman filter was first used as a promising linear model to be assessed against heavier non-linear machine learning models.
On the hand of machine learning models, Verbois et al. distinguished the models effective to reduce the RMSE, including multi-layer perceptron (MLP) and gradient boosting machine (GBM), and the models promising for reducing the MAE, notably the standard vector regression (SVR). Suksamosorn et al. also pointed out the effectiveness of the random forest (RF) model for a RMSE-optimization.
It's why I am going to compare the following models, against the reference raw forecasted irradiance.

- Kalman filter model (KF).

  The Kalman filter is a recursive estimator. This means that only the estimated state from the previous time step and the current measurement are needed to compute the estimate for the current state.

  The correction procedure involves two groups of equations: time update equations and measurements update equations, time update equations are responsible for making a first guess of the next solar irradiance prediction error, based on the last state of the measured error and error covariance estimates, obtaining an a priori prediction for the next time step; the measurement update equations will then incorporate new measurements into the first guess, obtaining improved a posteriori predictions.

  My understanding of the general Kalman filter was greatly thanks to Becker, and I practised the filter thanks to Labbe.

  In the context of irradiance forecasting, I followed the path from Suksamosorn et al..

- Gradient boosting machine model (GBM). GBM creates an ensemble of weak learners, meaning that it combines several smaller, simpler models in order to obtain a more accurate prediction than what an individual model would produce. Gradient boosting works by iteratively training the weak learners on gradient-based functions and incorporating them into the model as "boosted" participants. For more information, see notably Analytics for the theory and Bento (a) for the python practise.

- Support vector regression model (SVR). The standard vector regression method is often used in cases where there are multiple input variables, each of which may have an effect on the output variable. The goal is to find the best linear combination of these input variables to predict the output variable.

To estimate the coefficients of the linear function, standard vector regression uses a method called least squares regression. This involves finding the values of the coefficients that minimize the sum of the squared differences between the predicted and actual values. Here is an interesting article on the subject: Sharp.

- Random forest model (RF). The Random Forest algorithm is an ensemble method used for machine learning. It creates multiple decision trees, each trained on a different subset of data and considering random features for splitting. The final prediction is made by combining the predictions of these trees through voting (for classification) or averaging (for regression), resulting in improved accuracy and reduced overfitting. Again, here is a link for a hands-on practise of the RF algorithm: Bento (b).

- Multiple-layer perceptron model (MLP). The Multilayer Perceptron (MLP) is a type of artificial neural network used in machine learning. It consists of multiple layers of interconnected nodes (neurons) where each node computes a weighted sum of its inputs, passes it through an activation function, and then forwards the result to the next layer. MLPs are commonly used for various tasks such as classification, regression, and pattern recognition, and they can learn complex relationships in data. They can be trained using backpropagation, adjusting the weights between nodes to minimize the difference between predicted and actual outputs.

All this models we'll be compared during the internhip, and all the data wrangling architecture around it can be found either in the README or more specifically in the source code of my repo ACCOU.

2.4 Model performances evaluation strategy

The benchmarking consists in evaluating the performances on each metrics of each one of the model optimized with the corresponding metrics.
We assess the performances of the trained models on their performances on the test year.
The big picture will be given by a global significance matrix that will compare all the models performances regarding the particular metrics across the 4 sites.
This matrix will allow us to discern the most pertinent models for each of our study metrics.

Then, to verify that the models indeed perform well in the detail and across the different times of the day, we are going to plot scatter plots and data distributions of the MBE for each site.
This dual approach will ensure us that global results indeed translate into improved performances for each hour of the day.

3 Results and discussion

3.1 Post-processing a single forecasting NWP model

3.2 Showcase of the hybrid model

4 Conclusions and perspectives

4.1 Results summary

4.2 Suggestions for future improvements

4.3 My learnings from the internship

References

M. ACCOU. My intership gitlab repo. URL https://gitlab.soleka.org/maccou/maccou.

D. Analytics. Gradient boosting algorithm: Concepts, example. URL https://vitalflux.com/gradient-boosting-algorithm-concepts-example/?utm_content=cmp-true.

A. Becker. Kalmanfilter.net. URL https://www.kalmanfilter.net/default.aspx.

C. Bento. Gradient boosted decision trees explained with a real-life example and some python code, a. URL https://towardsdatascience.com/gradient-boosted-decision-trees-explained-with-a-real-life-example-and-some-python-code-77cee4ccf5e.

C. Bento. Random forests algorithm explained with a real-life example and some python code, b. URL https://towardsdatascience.com/random-forests-algorithm-explained-with-a-real-life-example-and-some-python-code-affbfa5a942c.

R. R. Labbe. Kalman and bayesian filters in python. URL https://github.com/rlabbe/Kalman-and-Bayesian-Filters-in-Python.

E. Lorenz, J. Remund, S. C. Müller, W. Traunmüller, G. Steinmaurer, D. Pozo, V. Lara, L. Ramirez, M. G. Romeo, C. Kurz, L. M. Pomares, and C. G. Guerrero. BENCHMARKING OF DIFFERENT APPROACHES TO FORECAST SOLAR IRRADIANCE.

M. J. Mayer and D. Yang. Calibration of deterministic NWP forecasts and its impact on verification. 39(2):981–991. ISSN 01692070. doi: 10.1016/j.ijforecast.2022.03.008. URL https://linkinghub.elsevier.com/retrieve/pii/S0169207022000486.

T. Sharp. An introduction to svr. URL https://towardsdatascience.com/an-introduction-to-support-vector-regression-svr-a3ebc1672c2.

S. Suksamosorn, N. Hoonchareon, and J. Songsiri. Post-processing of NWP forecasts using kalman filtering with operational constraints for day-ahead solar power forecasting in thailand. 9:105409–105423. ISSN 2169-3536. doi: 10.1109/ACCESS.2021.3099481. URL https://ieeexplore.ieee.org/document/9494359/.

H. Verbois, Y.-M. Saint-Drenan, A. Thiery, and P. Blanc. Statistical learning for NWP post-processing: A benchmark for solar irradiance forecasting. 238:132–149. ISSN 0038092X. doi: 10.1016/j.solener.2022.03.017. URL https://linkinghub.elsevier.com/retrieve/pii/S0038092X22001839.