Case Study of Wind Energy Production Predicted by ARIMA Model

Hyuntae Park

Department of Chemical and Biological Engineering, Korea University, 145 Anam-ro, Seongbuk-gu, Seoul, 02841, Republic of Korea.
parksapphire@korea.ac.kr

ABSTRACT

Wind power is one of the most efficient and reliable renewable energy resources. For the effective use of wind power, prediction of wind energy generation is an important task due to its stochastic nature. In this article, a one day-ahead forecast of Estonian wind energy production is made based on Autoregressive Integrated Moving Average (ARIMA) model with seasonality. Comparison with the actual production provided by an Estonian energy company, Elering showed high errors in the forecast. Explanations for the deviation and potential solutions are presented.

Commented [상곽1]: From what?

Keywords: Renewable wind energy, Wind power forecasting, ARIMA

INTRODUCTION

Wind energy production fluctuates seasonally and throughout the day with chaotic turbulence. Thus, accurate forecasting of wind power is highly important for designing and locating a stable and reliable power system based on wind energy [1]. Statistical approach is a class of forecasting method of wind power production. It is based on historical wind energy production data and involves the application of machine learning algorithms (i.e., neural network) such as ARIMA, Support Vector Machine (SVM), and Random Forest regression [2]. In this article, a case study of short-term (day-ahead) forecast is performed based on statistical approach as an extension from Shabbir et al.'s work [3]. Especially, the scope of model selection was expanded by applying ARIMA to forecast the next day's wind energy production. An evaluation is done for the accuracy of the selected method.

The ARIMA model is consisted of three parts of autoregressive (AR), moving average (MA), and integrated (I). The AR part is used to describe the current value of a time-series, x_t, as a function of p past values. Since the AR model assumes the linear dependence of current value from past values, x_t is expressed as a linear combination of order p as follows [4],

Commented [상곽2]: 참고문헌 순서 사용 조심!

$$x_t = \sum_{i=1}^{p} \varphi_i \, x_{t-i} + \varepsilon_t \quad (1)$$

where φ_i are the parameters of the AR model and ε_t denotes the white noise error term. The MA model is an alternative to the AR model, linearly relating x_t to q past error terms, as shown below,

$$x_t = \varepsilon_t + \sum_{i=1}^{q} \theta_i \, \varepsilon_{t-i} \quad (2)$$

where θ_i are the parameters and ε_t and ε_{t-i} represent current and past error terms, respectively. Finally, the I part notates the differencing degree. Both the AR and MA models are applied to stationary data, where the mean μ and autocovariance γ is invariant for any time point t. However, some time series data show change in μ, indicating an inherent trend, or exhibit a seasonal cycle, both of which are considered non-stationary. Thus, differencing is implemented to delete the trend or seasonality in the data, and approximate it as stationary, which is done as follows,

Commented [상곽3]: What trend? Be more specific.

$$y_t = x_t - x_{t-1} \quad (3)$$

By combining three parts into one, the ARIMA model can be expressed as ARIMA(p, d, q). For data with a seasonal cycle, seasonal orders *P, D,* and *Q* are additionally introduced with the seasonal period *m*. Then, the final form is expressed as SARIMA(p, d, q)(P, D, Q)[m].

RESULTS AND DISCUSSION

The case study of forecasting of wind power production was performed based on the data provided through Elering's website [5]. The wind power generated in Estonia for 30 days in September 2023 is shown in Fig. 1(a) in comparison with the company's prediction. To analyze the inherent pattern of the data, decomposition with respect to trend and seasonality was done by using the seasonal_decompose function from statsmodels.tsa.seasonal python library [6]. The function assumes a provided dataset can be decomposed into the linear sum of a trend component, a seasonal component, and a residual component. The trend component is analyzed first by tracking the change of the mean value μ based on the MA model. Then, seasonality in the data is inspected by identifying periodic patterns within the given time series unit. The two analyzed components are subtracted from the original data to obtain the residual component. The result of seasonal decomposition applied to Estonian wind power data is shown in Fig. 1(b). The 'Trend' curve reveals an irregular trend in the data, while the 'Seasonal' curve shows periodicity on a daily

Commented [상곽4]: , which is…? 어떻게 구하게 되는지를 설명. 계절별 데이터를 뭐 어떻게 한것인지?

basis. Considering that the time series was given hourly, the period can be set to 24 hours, suggesting a m value of 24. Moreover, the autocorrelation plot of the original data in Fig. 1(c) steadily decreases, implying non-stationarity. Applying first-order differencing results in the autocorrelation plot exponentially decreasing to oscillate within a relatively narrow range, indicating that stationarity is reached. Since stationarity is achieved with a single degree of differencing, further operations are unnecessary, and the optimal values for d and D are set to 1.

Commented [상곽5]: Why?

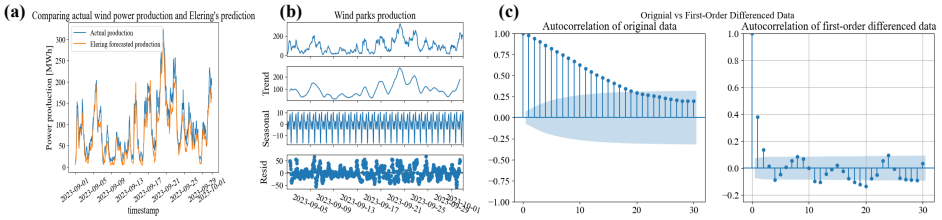

Figure 1. Plots of the data analysis. (a) Actual wind power production data from Elerning compared to its prediction, (b) Decomposition of wind power production data in terms of trend, seasonal, and residual components, (c) Comparison of the autocorrelation plots of original wind power production data and the data after applying first-order differencing.

To determine optimal values for the remaining parameters, a best-fitted model is selected among those generated with different parameter values based on performance evaluation. In this case study, Akaike information criterion (AIC) was used as the evaluation criteria. The AIC score for a model is calculated as (4), where k is the number of parameters in the model and $\hat{L}$ is the maximized likelihood function for the model [8].

$$\mathrm{AIC} = 2\mathrm{k} - 2\ln(\hat{L}) \tag{4}$$

For a SARIMA model, k is calculated by the sum of p, d, q, P, D, Q, m plus 1 for the constant term. Moreover, the Gaussian distribution was used as the likelihood function, based on the assumption that the estimation errors are normally distributed. The function auto_arima from the python library pmdarima.arima [7], searches for the model with the lowest AIC score within a given parameter range. The range of examination considered for fitting the Estonian wind power production data were [1, 5] for p and q, [1, 3] for P and Q, and [1, 2] for d and D. Additionally, three types of trends were set as a parameter for the auto_arima function. As mentioned previously, the 'Trend' component of Fig. 1(b) indicates that the given data exhibits stochastic trend. Since it

cannot be specified as a certain type, approximation is required during the ARIMA fitting process. First, approximation to a constant trend was applied, where having a constant trend implies the μ value of the data is shifted and maintained at a constant value. The optimal model suggested for a constant trend approximation was SARIMA(1, 1, 0)(3, 1, 0)[24] with an AIC score of 5488.5696. Also, the trend was estimated as linear. Data with linear trend have linearly increasing μ values with time. Such estimation resulted with an optimal model of SARIMA(2, 1, 2)(0, 1, 1)[24] with 5534.6325 as its AIC score. Finally, an assumption that the trend is constant and linear was applied. Such trend is exhibited when μ is shifted to a certain intercept value and linearly increased. The best model with for this case was SARIMA(1, 1, 1)(0, 1, 1)[24] with 5539.6935 AIC score.

The SARIMAX function from the statsmodels.tsa.statespace.sarimax python library [6] was applied to train each of the optimal SARIMA models with the wind power production data of September. The trailing X of SARIMAX stands for 'exogenous variable', which is neglected in this case study. Then, predictions were made from the trained model for 24 hours ahead and is compared with the actual production of October and the forecast made by Elering. The results presented in Fig. 2. Fig. 2(b) show all three ARIMA forecasts with different trend assumptions described previously. Among the three, linear assumption fits best with a RMSE value of 52.3604. Constant and linear trend assumption fits worst with a RMSE value of 88.9175, and simple constant trend shows intermediate fitting with 71.1661 RMSE.

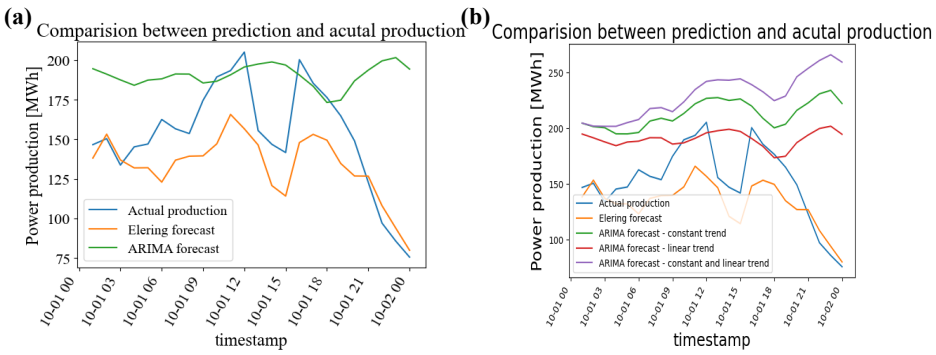

Figure 2. Plots of the forecast. (a) Prediction based on linear trend approximation compared with actual wind power production and Elering's forecast, (b) Predictions of different trend approximations.

The best fit with linear trend is presented solely in Fig. 2(a) for analysis. As seen in the plot, SARIMA model's prediction overestimates the power production and seems to weakly follow the tendency of the actual production until 10-01 18:00. Afterwards, however, it shows deviation in direction from the actual production data.

Some additional methods were considered to improve the prediction. Since the bold approximation of stochastic trend to linearity is regarded as a major cause for the divergence of prediction, a polynomial fit for the trend was attempted. However, due to the high randomness, even a polynomial of degree up to 7 was not able to fit the trend.

Thus, for further improvement, introducing an exogenous variable is an option. The X in SARIMAX denotes an exogenous variable, which influences the dependent variable, but is independent of it. For this case study, wind speed data can serve as an exogenous variable for the dependent variable, wind power production, as it is known to impact wind production rates [1][2]. Therefore, adding wind speed data corresponding to the time span of interest can lead to a better forecast.

Furthermore, it is speculated that the addition of data preprocessing and model validation stages could enhance the accuracy. Finally, instead of making forecast for 24-time stamps at once, a step-by-step prediction could decrease the error.

Commented [상곽6]: In 5 page report, do this.

The Jupyter Notebook and data used for this case study can be accessed in [9].

CONCLUSION

In this article, a statistical approach based on the ARIMA model is used to make forecasts of wind power production. An optimal model of SARIMA(2, 1, 2)(0, 1, 1)[24] was selected and trained. The model assumed linear increase in the mean value over time, resulting in a prediction with an RMSE value of 52.3604. Notably, this is larger than that of Elering's algorithm (i.e., 26.1859) and Shabbir et al. [3]'s prediction with SVM (i.e., 18.481). The suggested reasons for this deviation include crude approximation of the trend and absence of data preprocessing and model validation steps. To enhance accuracy, potential improvements can be made by refining such factors, introducing an exogenous data set of wind speed, and adopting a step-by-step prediction approach.

REFERENCE

[1] Demolli, H., Dokuz, A. S., Ecemis, A., & Gokcek, M. (2019). Wind power forecasting based on daily wind speed data using machine learning algorithms. *Energy Conversion and Management, 198*.

[2] Wang, X., Guo, P., & Huang, X. (2011). A Review of Wind Power Forecasting Models. *Energy Procedia*.

[3] N. Shabbir, R. AhmadiAhangar, L. Kütt, M. N. Iqbal and A. Rosin, "Forecasting Short Term Wind Energy Generation using Machine Learning," *2019 IEEE 60th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON)*, Riga, Latvia, 2019, pp. 1-4.

[4] Shumway, R. H., & Stoffer, D. S. (2017). *Time Series Analysis and Its Applications: With R Examples.* Springer.

[5] Elering AS. *Wind parks*. Retrieved from Elering Live: https://dashboard.elering.ee/en/system/with-plan/production-renewable?interval=minute&period=days&start=2023-10-08T15:00:00.000Z&end=2023-10-09T14:59:59.999Z

[6] Seabold, Skipper, and Josef Perktold. "statsmodels: Econometric and statistical modeling with python." *Proceedings of the 9th Python in Science Conference.* 2010.

[7] Smith, Taylor G., *et al*. pmdarima: ARIMA estimators for Python, 2017-, http://www.alkaline-ml.com/pmdarima [Online; accessed 2023-10-25]

[8] Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control, 19(6), 716-723.*

[9] Hyuntae, P. (2023). enereng2023. GitHub. https://github.com/HyuntaeKR/enereng2023