Journal of Computer Science and Technology Studies
ISSN: 2709-104X
DOI: 10.32996/jcsts
Journal Homepage: www.al-kindipublisher.com/index.php/jcsts

JCSTS
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

| RESEARCH ARTICLE

Comparative Machine Learning and Time Series Forecasting of Wind Power Output using SCADA Data

Md Thouhid Ul Alam[1] ✉ and Abu Sayeed Mozomder[2]
[1]Master of Accountancy & Data Analytics, The University of Mississippi
[2]Master in Economics and Business Administration, Major in Business Analytics, Norwegian School of Economics
Corresponding Author: Md Thouhid Ul Alam, **E-mail**: thouhid.ua@gmail.com

| ABSTRACT
The rapid growth of wind energy as a renewable power source has presented both opportunities and operational challenges due to its inherent intermittency and non-linearity. Accurate short-term forecasting of wind power output is vital for improving grid reliability, reducing operating costs, and enhancing energy market efficiency. This study presents a comprehensive comparative analysis of traditional time series models and modern machine learning algorithms for short-term wind power forecasting. Using high-frequency SCADA data from a commercial wind turbine in Turkey, we evaluate the predictive performance of nine models: Seasonal Naive, Exponential Smoothing (ETS), ARIMA, ARIMAX, Dynamic Harmonic Regression, Linear Regression, Gradient Boosted Trees (GBM), and Generalized Additive Models (GAM). Root Mean Square Error (RMSE) is employed as the primary accuracy metric on a hold-out test set. Among the models analyzed, the Gradient Boosted Tree model demonstrated superior performance with the lowest RMSE of 16.05, followed by ARIMAX (RMSE = 20.8), GAM (RMSE = 29.7), and linear regression using external meteorological inputs (RMSE = 29.9). Traditional statistical models such as ETS and ARIMA showed comparatively lower performance, particularly in handling non-linear patterns and multiple seasonalities. The study highlights the importance of integrating exogenous variables—such as wind speed, direction, and theoretical power curve—into forecasting frameworks to capture complex relationships and improve accuracy. Our findings emphasize that ensemble learning methods and hybrid statistical-ML approaches offer meaningful advancements in renewable energy forecasting. These insights provide valuable guidance for energy planners, system operators, and stakeholders seeking to optimize renewable integration and grid stability. Future work can explore deep learning architectures and spatiotemporal models across multi-turbine datasets for broader applicability.

| KEYWORDS

Machine Learning, Time Series, Wind Power Forecasting, SCADA, Output Prediction

| ARTICLE INFORMATION

ACCEPTED: 01 June 2025 **PUBLISHED:** 30 June 2025 **DOI:** 10.32996/jcsts.2025.7.7.2

1. Introduction

The integration of renewable energy sources into power systems has become a cornerstone in achieving global sustainability goals, with wind energy emerging as one of the most promising and rapidly expanding technologies. According to the International Energy Agency (IEA, 2023), wind power accounted for over 15% of global renewable electricity generation in 2022 and is projected to play a critical role in achieving net-zero targets. However, the intermittent and stochastic nature of wind presents significant challenges to grid reliability, power market stability, and energy storage planning (Zhang et al., 2014; Foley et al., 2012).

Accurate short-term forecasting of wind power output is essential for mitigating these challenges. Grid operators rely on these forecasts to balance supply and demand, reduce operating reserves, and avoid penalties arising from imbalances in the

electricity market (Giebel et al., 2011). Forecasting also supports economic dispatch, maintenance scheduling, and renewable portfolio optimization (Soman et al., 2010). While traditional statistical models such as ARIMA and Exponential Smoothing have long been applied to this domain, they often fall short in capturing the non-linear relationships and high-frequency variations inherent in wind data (Zhou et al., 2011).

Recent advances in machine learning (ML) have led to a paradigm shift in time series forecasting for renewable energy applications. Models such as Gradient Boosted Trees, Random Forests, and Deep Neural Networks have demonstrated superior performance in capturing complex, non-linear dependencies between meteorological variables and power output (Rashid et al., 2020; Singh et al., 2021). In particular, ensemble methods and hybrid models combining time series and regression components—such as ARIMAX and dynamic harmonic regression—offer promising accuracy improvements over purely statistical or ML-based approaches (Wang et al., 2017).

This study contributes to the growing literature by systematically comparing a range of forecasting and predictive modeling techniques using SCADA-derived data from a single wind turbine. We evaluate models including Seasonal Naive, ETS, ARIMA, ARIMAX, Linear Regression, Boosted Trees, Generalized Additive Models (GAM), and Dynamic Harmonic Regression. The comparison is based on Root Mean Square Error (RMSE) over a hold-out test set, offering practical insights into model suitability under multiple seasonalities and noisy data environments. Our findings underscore the value of ensemble learning approaches, particularly Boosted Trees, in delivering robust and accurate wind power predictions for real-world operational use.

2. Literature Review

Short-term wind power forecasting has received considerable attention in recent decades due to the growing integration of intermittent renewable energy sources into modern power systems. Accurate forecasting techniques enable grid operators and market participants to enhance operational efficiency, reduce balancing costs, and improve the economic dispatch of electricity (Zhang et al., 2014; Soman et al., 2010).

Traditional approaches to wind power forecasting have often relied on statistical time series methods, such as the Autoregressive Integrated Moving Average (ARIMA) model, which are well-suited for stationary data but may fall short in capturing the non-linear dynamics and multiple seasonalities observed in real-world wind data (Box et al., 2015; Hyndman & Athanasopoulos, 2021). Exponential Smoothing State Space Models (ETS) have also been used to model level, trend, and seasonal components, though their performance may degrade in the presence of non-Gaussian noise or complex temporal dependencies (Hyndman et al., 2008).

In an effort to improve forecasting accuracy, researchers have increasingly turned to **machine learning (ML) techniques**, which are capable of modeling non-linear and high-dimensional relationships without explicit assumptions about data distributions. For instance, **Random Forest** and **Gradient Boosting Machines (GBMs)** have shown strong predictive performance in wind power forecasting tasks by leveraging ensemble learning and residual fitting (Natekin & Knoll, 2013). Rashid et al. (2020) used Random Forests on SCADA data to predict the capacity factor and achieved mean absolute errors below 4% in certain configurations.

More recently, **hybrid and deep learning models** have emerged as promising alternatives. Singh et al. (2021) compared Random Forest, Long Short-Term Memory (LSTM), and Extreme Gradient Boosting (XGBoost) models on wind farm SCADA datasets and found that ensemble models outperformed classical neural networks in forecasting accuracy. These models also benefit from their ability to integrate exogenous variables such as wind speed, direction, and theoretical power curves, making them well-suited for real-time applications (Wang et al., 2017).

Additionally, **Generalized Additive Models (GAM)** have been applied in cases where smooth non-linear effects are hypothesized between predictors and response variables (Hastie & Tibshirani, 1990). These models offer an interpretable alternative to black-box ML methods while allowing flexible modeling of non-linearity through spline functions.

The literature also highlights the growing role of **regression models with ARIMA errors (ARIMAX)**, which combine the strengths of regression and time series modeling to incorporate external predictors and autocorrelated residuals. Studies such as those by Xie et al. (2014) and Gao et al. (2019) demonstrate how such hybrid models improve upon pure ARIMA or ML-based techniques, especially when handling seasonally adjusted datasets.

Despite these advancements, a consensus remains elusive regarding the best forecasting method under all scenarios. Factors such as time resolution, geographical variability, turbine technology, and meteorological inputs greatly influence model

performance (Foley et al., 2012). This underscores the importance of comparative model studies using consistent datasets and evaluation metrics, which this paper seeks to address.

3 Methodological Framework

This study employs both classical statistical time series forecasting methods and modern machine learning techniques to compare predictive performance in short-term wind power output forecasting. The analysis is based on SCADA data from a wind turbine in Turkey, covering hourly measurements for the year 2018. The methodology consists of data preprocessing, model development, and performance evaluation based on Root Mean Square Error (RMSE).

3.1 Data Source

The dataset used in this study was collected from a **Supervisory Control and Data Acquisition (SCADA) system** linked to a single commercial wind turbine located in Turkey. SCADA systems are widely adopted in wind energy operations and provide high-frequency time series data on turbine performance and environmental conditions, including:

- **Wind Speed** (in meters per second),
- **Wind Direction** (in degrees),
- **Theoretical Power Curve Output** (in kilowatts), and
- **Actual Power Output** (in kilowatts).

The dataset spans a full calendar year, from **January 1, 2018, to December 31, 2018**, with an original temporal resolution of **10-minute intervals**. After initial preprocessing, the data were aggregated to **hourly intervals** to better align with standard forecasting horizons and grid operation schedules. This aggregation also reduces short-term volatility and sensor noise inherent in high-frequency data (Soman et al., 2010).

A total of **8,439 hourly observations** were retained after filtering for completeness and quality. Missing values were not present; however, some observations were found to have negative or zero actual power output. These were retained after inspection, as they likely reflect real turbine idling or calm wind conditions.

Due to data-sharing constraints, the dataset is **not publicly available** but was obtained through an academic-industry collaboration for research purposes. However, similar open-access datasets are available for benchmarking, such as:

- **NREL WIND Toolkit** (National Renewable Energy Laboratory):
  https://www.nrel.gov/grid/wind-toolkit.html
- **GEFCom2012/GEFCom2014 Wind Forecasting Challenge Datasets**:
  https://www.drhongtao.com/gefcom

These public datasets may be used for validation or replication of similar studies.

3.2 Data Preprocessing

The original data was collected at 10-minute intervals and included four variables: wind speed (m/s), wind direction (degrees), theoretical power curve (kW), and actual power output (kW). To reduce noise and align with operational decision-making time frames, the data was aggregated to **hourly intervals** using summation for power output and averaging for other variables, yielding 8,439 hourly observations.

To stabilize variance and approximate normality, we applied a **Box-Cox transformation** to the response variable:

$$y^{(\lambda)} = \begin{cases} \frac{y^{\lambda}-1}{\lambda}, & \text{if } \lambda \neq 0 \\ \ln(y), & \text{if } \lambda = 0 \end{cases}$$

A λ value of approximately 0.386 was estimated using maximum likelihood (Box & Cox, 1964). The transformed series was then differenced if needed to achieve stationarity, assessed via **Augmented Dickey-Fuller (ADF)** tests and **ACF/PACF** plots.

3.3 Forecasting Methods

3.3.1 Seasonal Naive and ETS

The **Seasonal Naive (SNAIVE)** method assumes that the future value at a given time is equal to the value at the same time in the previous season. This serves as a simple benchmark model.

The **ETS** model, defined in state-space form, considers error, trend, and seasonality components (Hyndman et al., 2008). The general form is:

$$y_t = l_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t$$

$$l_t = l_{t-1} + b_{t-1} + \alpha\varepsilon_t, \quad b_t = b_{t-1} + \beta\varepsilon_t, \quad s_t = s_{t-m} + \gamma\varepsilon_t$$

where:

- l_t: level,
- b_t: trend,
- s_t: seasonality,
- α, β, γ: smoothing parameters.

We selected between ETS(A,N,A), ETS(A,A,A), and Holt-Winters models using the AICc criterion.

3.3.2 ARIMA and ARIMAX

The **ARIMA(p,d,q) (P,D,Q)[m]** model captures autocorrelations and seasonality in stationary data (Box et al., 2015). The model can be expressed as:

$$\Phi_p(B)\Phi_P(B^m)(1-B)^d(1-B^m)^D y_t = \Theta_q(B)\Theta_Q(B^m)\varepsilon_t$$

Where:

- B is the backshift operator,
- Φ, Θ are polynomials of orders p, P, q, Q,
- m is the seasonal frequency (e.g., 24 for daily seasonality in hourly data).

To incorporate exogenous variables, we used **ARIMAX**, extending ARIMA as:

$$y_t = \beta_0 + \sum_{i=1}^{k} \beta_i x_{i,t} + z_t, \quad z_t \sim ARIMA(p,d,q)$$

Where $x_{i,t}$ are the regressors: wind speed, theoretical power, and wind direction.

3.3.3 Dynamic Harmonic Regression

To capture long-period seasonalities, **Dynamic Harmonic Regression (DHR)** incorporates Fourier terms as exogenous regressors:

$$y_t = \beta_0 + \sum_{k=1}^{K} \left[\alpha_k \cos\left(\frac{2\pi k t}{m}\right) + \gamma_k \sin\left(\frac{2\pi k t}{m}\right) \right] + \varepsilon_t$$

Here, KKK controls the number of harmonics, and mmm represents the seasonal period (e.g., 24 for daily, 168 for weekly). DHR effectively approximates multiple seasonalities (Taylor et al., 2017).

3.4 Predictive Modeling Techniques

3.4.1 Linear Regression

Two **Ordinary Least Squares (OLS)** models were fitted:

- Model A: Power output ~ time-based features (trend, seasonality)
- Model B: Power output ~ wind speed, theoretical power, wind direction

The standard OLS form:

$$\hat{y} = \beta_0 + \sum_{i=1}^{n} \beta_i x_i + \varepsilon$$

Model performance was evaluated via **Adjusted R^2** and **RMSE**.

3.4.2 Gradient Boosted Trees (GBM)

We implemented GBMs using the **gbm** package (Ridgeway, 2006). Boosting iteratively fits weak learners to the residuals of previous models:

$$F_m(x) = F_{m-1}(x) + \nu h_m(x)$$

Where:

- F_m is the boosted model at iteration mmm,
- $h_m(x)$ is the base learner (typically a decision tree),
- ν is the learning rate.

Hyperparameters included:

- n.trees = 5000
- interaction.depth = 4
- shrinkage = 0.01

Feature importance was derived from relative influence measures.

3.4.3 Generalized Additive Models (GAMs)

GAMs model the response as a sum of smooth functions:

$$y_t = \beta_0 + f_1(x_1) + f_2(x_2) + \cdots + f_n(x_n) + \varepsilon_t$$

Here, f_i are spline-based smoothers (e.g., cubic regression splines). We used the **mgcv** package (Wood, 2017) with gam () and compared three GAM formulations with different degrees of smoothness.

3.4 Model Evaluation

All models were trained on data from **January 1 – December 24, 2018**, and tested on **December 25–31, 2018**. The **Root Mean Square Error (RMSE)** was used as the primary evaluation metric:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}$$

Models were ranked based on test set RMSE.

3.5 Data Features

3.5.1 Data

Wind farms are typically linked to a Supervisory Control and Data Acquisition (SCADA) system, which constantly gathers data from individual wind turbines. The logged information encompasses crucial parameters like power production, wind direction, and wind speed [3]. Leveraging the predictive power of wind turbine forecasts allows wind farm management teams to make informed decisions pertaining to power production, consumption, and storage capacity management within smart grids.

For the purpose of this thesis, we applied the data generated by a wind turbine in Turkey from January 1st, 2018, to December 31st 2018. We aim to use this data to provide forecasts for six days by dividing the data into a training set, from January 1st, 2018, to December 24st 2018 and a test set, December 25st 2018, to December 31st 2018.

3.5.2 Data Transformation and Descriptive Statistics

Our initial data is composed of four parts, the wind speed and wind direction that the turbine uses for electricity generation, the theoretical power given the wind speed for that moment, and the actual power generated by the turbine. The data is for 10 minutes intervals, with no mission value but have aberrant values. The data are quite clean and only need simple adjustments. Aggregating 10-minute data to hourly data for wind power forecasting is done primarily to reduce the effects of short-term variability and noise, thereby enhancing the stability and reliability of the forecasting model. This simplifies the data structure and aligns with typical operational and market decision-making periods, which are often based on hourly schedules. After aggregation, we have 8439 observations with 5 variables. The variables include Date, Power_kw (actual power generated by the turbine), Windspeed, Theoretical_Power_Curve (theoretical power given the wind speed), wind direction. We have used this code to aggregate data:

We have summed the actual power production, and for aggregating other columns we have used the mean function. As it is logical to have total hourly production, and for other means is feasible.

We compute some descriptive statistics for actual power production/Power_kw.

Min.	1st Qu.	Median	Mean	3rd Qu.	.Max.
-1.077	468.285	5061.27	0 7829.990	14638.404	21626.460

Table 1: Some descriptive statistics for actual power production/Power_kw

From the summary table we can see that the maximum power production is 21626.460kW and the minimum value is -1.077. When we investigate this further, we find that 1495 observations are zero, which is sensible because there are times that has no wind and no electricity production. Data appears to have a wide range, including negative and zero values, as well as a substantial variance. As data has many negative and zero values as well, we choose to use box-cox transformation instead of log transformation.

We get a lambda value of 0.3858352, and we transform the data before dividing the data set into train and test. After that we tried to use some time plots to find out trends, and seasonality in the data. From the two time series plots, it is hard to identify any seasonality and trend, because there are a lot of close values. So we have decided that we will examine autocorrelation coefficients with the ACF plot to see whether our data has trends and seasonality.

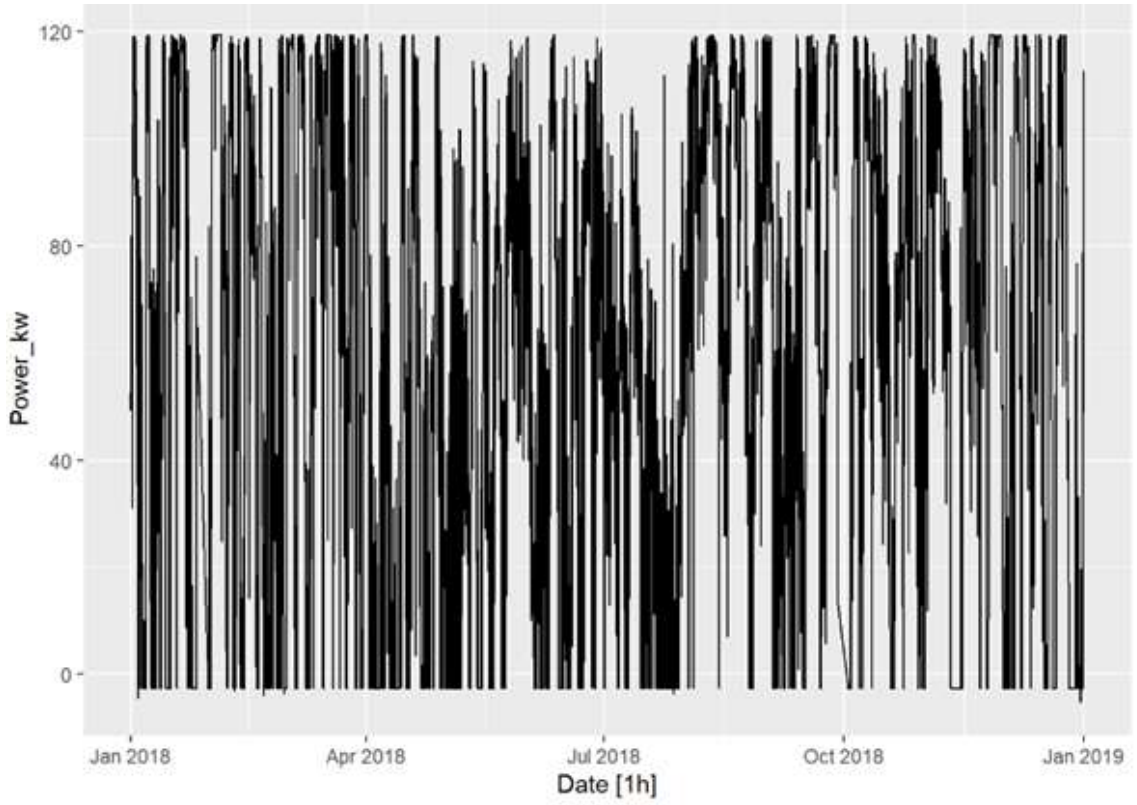


Figure 1: Time series plot to see hourly and daily seasonality

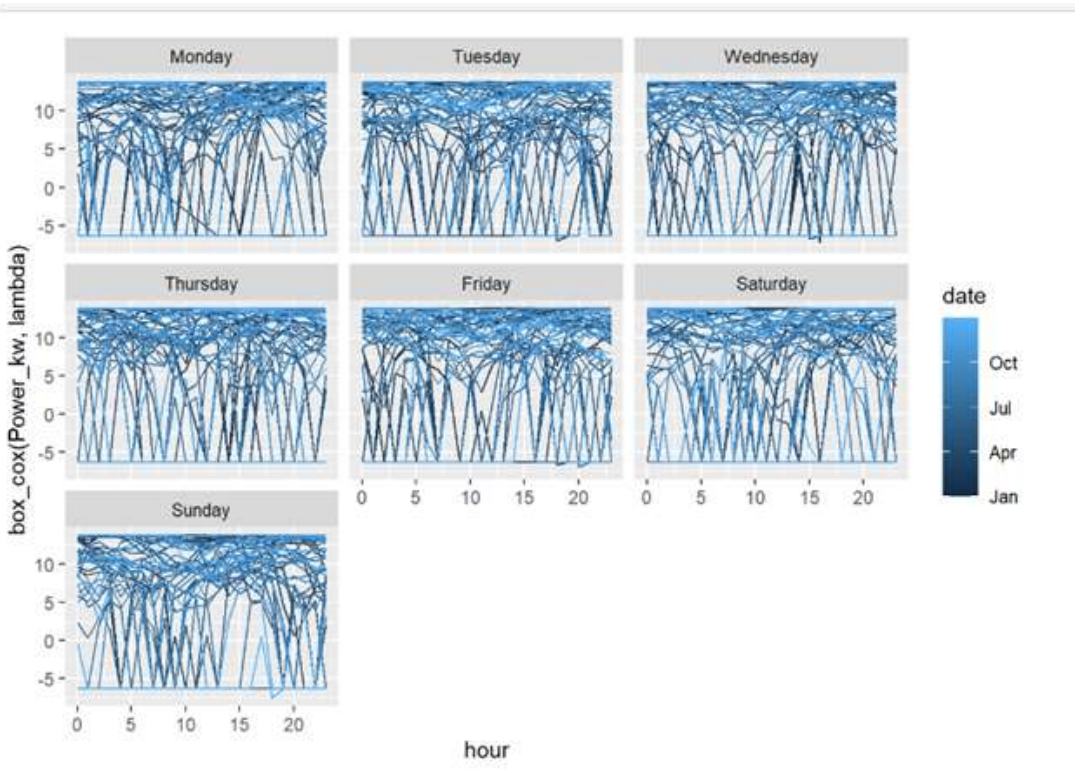


Figure2: Customize time series plots to see daily and weekly seasonality

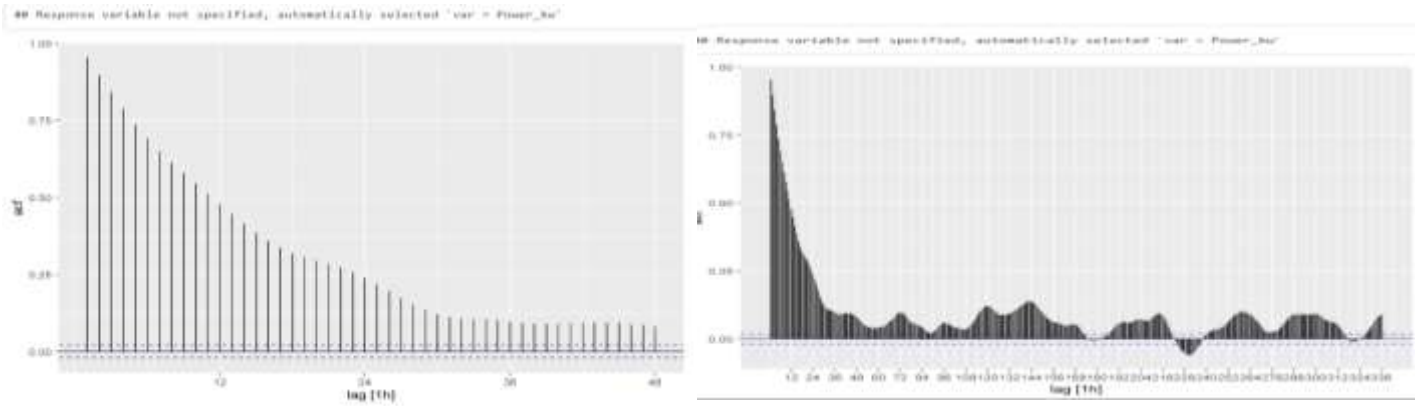


Figure 3: ACF plot of actual power production/Power_kw

From the ACF plot, it is clear that wind power production has both daily and weekly seasonality with strong trend, as slow decrease in the ACF as the lags increase is due to the trend, while the "scalloped" shape is due to the seasonality

4 Data Analysis and Modelling with Forecasting Methods

4.1 Seasonal NAIVE

As our dataset has both seasonality and trend, we have used the seasonal naive method as a benchmark forecasting method. The SNAIVE method is a simple forecasting approach, which assumes that the seasonal patterns will repeat exactly as in the past.

Model	Type	ME	RMSE	MAE	MPE	MAPE	MASE	RMSSE	ACF1
NAIVE (Power_	Test	-89.4	99.4	92.8	−Inf	Inf	NaN	NaN	0.634

Table 2: Accuracy Metrics of Seasonal Naive (SNAIVE) Model on Test Set

The RMSE (Root Mean Square Error) value for the test set, calculated using the accuracy() function, is 99.4. This value indicates the standard deviation of the residuals (prediction errors). Residuals are a measure of how far from the regression line data points are, so a lower RMSE is generally preferable as it would indicate that the errors made by the prediction are smaller. Thus, a lower RMSE value is generally indicative of a better fit to the data. We are going to use RMSE as our accuracy to compare different machine learning models. It's worth noting that some of the metrics are infinite or NaN in our results (MPE, MAPE, MASE, RMSSE). This could be due to division by zero, which typically indicates that there are zero values in our test dataset, and these metrics are calculated using divisions.

4.2 Exponential Smoothing (ETS)

When we use ETS method, we can choose from several types of ETS model such as additive model, multiplicative, Holt-Winters method, which uses multiplicative errors, an additive damped trend, and multiplicative seasonality, and function ETS() that automatically select the best model. We have used all of them as our data has multiple seasonalities and unsure about whether trends are constant.

Model	AICc
hw	123283
ets_auto	127142
additive	127150
multiplicative	145368

Table 3: Comparison of ETS Model Variants Using AIC

We have trained four different models (three ETS variants and one auto ETS model) on your training data and compared their AIC (Akaike Information Criterion) values. The AIC is a measure used to compare models. The lower the AIC, the better the model is considered to be, as a lower AIC suggests a model that has a better fit with a smaller number of parameters. The output suggests that the hw model, which is the Holt-Winters method with multiplicative errors, an additive damped trend, and multiplicative seasonality, is the best among the four models, with the lowest AIC of 123283. The second-best model according to the AIC is the ets_auto model, which is the automatic ETS model that selects the best model based on the AICc score. The additive and multiplicative models come in third and fourth place, respectively, according to their AIC scores.

Model	RMSE
ets_auto	102
additive	105
hw	102039
multiplicative	39136394

Table 4: Forecast Accuracy of ETS Models on Test Set Using RMSE

However, if you look at RMSE, it appears that the ets_auto model is doing better than any other models. According to AICc, how model was doing better. We are going to keep ats_auto model here as the best model, because it is doing better in an unknown data set. Otherwise, we can overfit our forecasting. It is also worth noting that in terms of the accuracy measurement in unknown dataset SNAIVE method is doing better than ets_auto model with RMSE of 99.4 (SNAIVE). As we said before that sometimes some simple models work better than other models, this is the proof. Apart from that this dataset has multiple seasonalities, and the ETS model is not good at handling multiple seasonality with long seasonal periods.

Parameter	Value
alpha	0.7383753
beta	0.0001096895
gamma	0.0001000083
phi	0.9699994

Table 5: Parameter Estimates and Structure of Selected ETS(A,Ad,A) Model

If we look at the report of the automatically generated ETS model, we can see the type of ETS model used. ETS(A,Ad,A) represents an ETS model with additive error, additive damped trend, and additive seasonality.These are the estimated parameters of the model, alpha (for level), beta (for trend), gamma (for seasonality), and phi (for dampening). These parameters determine how fast the model responds to changes in the level, trend, and seasonal pattern of the series.

4.3 ARIMA Model

Prior to fitting an ARIMA model, it is essential to ensure that the time series data is appropriately transformed and rendered stationary. In our case, the original series exhibited both a trend and seasonality, indicating non-stationarity.

To assess the degree of differencing required, we applied unit root tests. Specifically, the KPSS (Kwiatkowski–Phillips–Schmidt–Shin) test indicated a test statistic of 1.03 with a p-value of 0.01. This result suggests the presence of a unit root, thereby rejecting the null hypothesis of stationarity. Further evaluation using the ndiffs() function indicated that a first-order differencing was necessary to achieve stationarity in the trend component.

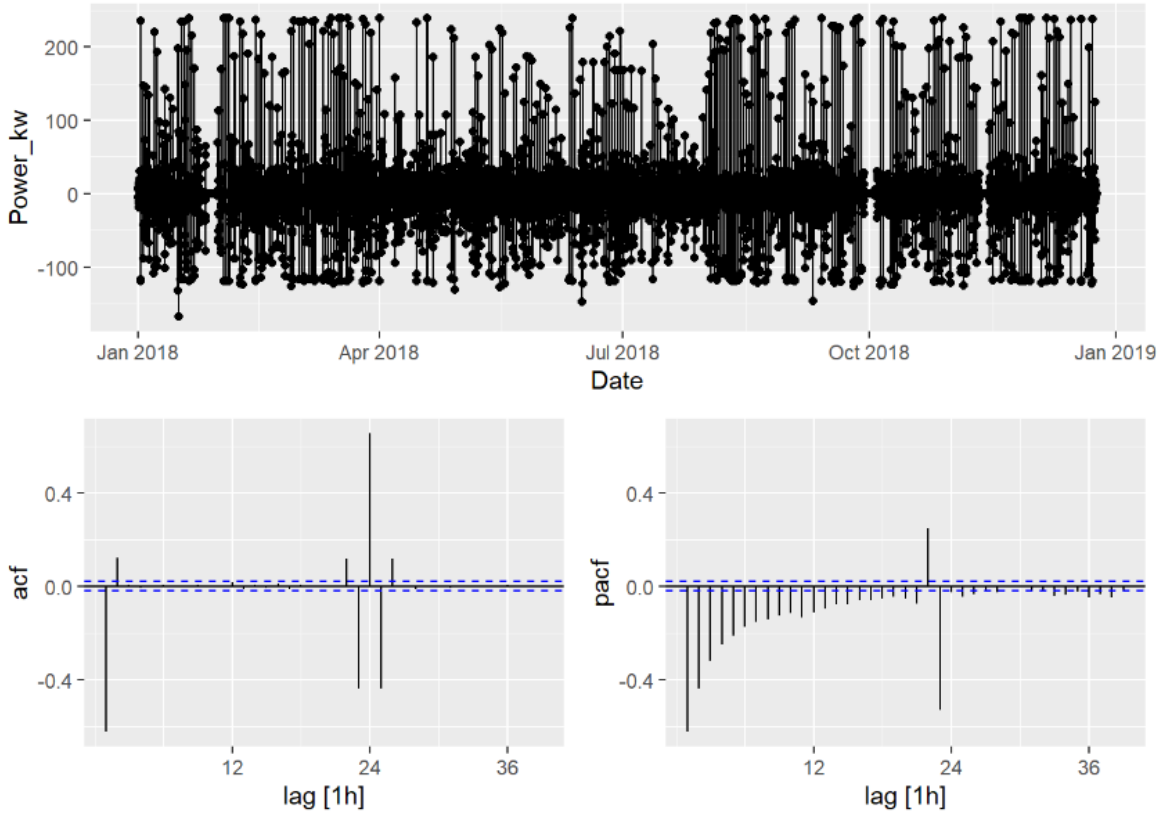
Given the seasonal pattern in the data, seasonal differencing was also considered. The need for both non-seasonal and seasonal differencing aligns with the visual evidence of trend and periodicity in the series.

Subsequently, we applied the required differencing to the data. We also utilized autocorrelation function (ACF) and partial autocorrelation function (PACF) plots—generated using gg_tsdisplay—to facilitate identification of appropriate ARIMA model

parameters. These diagnostics are crucial in determining the autoregressive (AR), moving average (MA), and seasonal components of the model.

In the above code, we have also used the gg_tsdisplay function to plot both ACF and PACF plots. Those are useful in estimating parameters in seasonal and non-seasonal ARIMA models.

Figure 4: ACF and PACF plots



If we look at ACF and PACF, especially in PACF plots, we can see that there are many significant spikes in lag. Therefore, it is a bit inconvenient to choose p or d randomly. Instead of choosing those randomly, we have used the ARIMA function, an automated algorithm.

Test	Value
KPSS Statistic	1.03
KPSS p-value	0.01
Number of Differences	1.0

Table 6: ARIMA Model Configuration and Forecast Accuracy

To make it work better, we choose to do stepwise = False. The fitted model is an ARIMA(0,1,4)(2,0,0)[24] model. This suggests that the time series data was differenced once to achieve stationarity (indicated by the '1' in the order of differencing in the non-seasonal part), and it uses 4 lagged error terms (moving average terms) in the non-seasonal part of the model (indicated by the '4'). For the seasonal part, the model uses 2 autoregressive terms (indicated by the '2'). The number '24' within brackets indicates

that the seasonal period of the series is 24. We can see from forecasting and RMSE that ARIMA model is not doing better than SANIVE or ETS model.

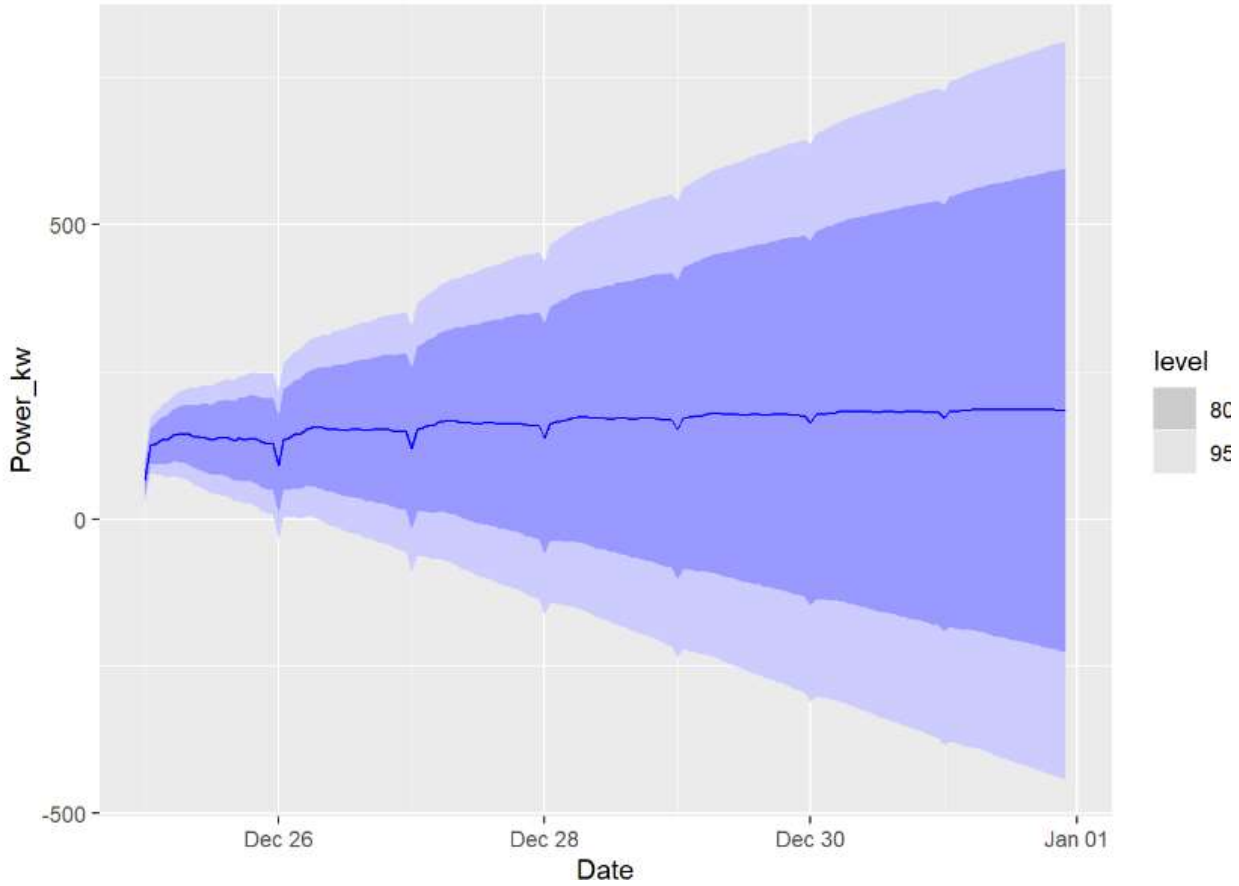


Figure 5: ARIMA Forecast Plot with 80% and 95% Confidence Intervals for Power Output

Model	Type	ME	RMSE	MAE	MPE	MAPE	MASE	RMSSE	ACF1
auto	Test	-147	152	147	−Inf	Inf	NaN	NaN	0.912

Table 7: Forecast Accuracy Metrics of Auto ARIMA Model on Test Set

4.4 Regression with ARIMA errors (ARIMAX)

As we can see, so far, the seasonal naive method is doing better than other models. We have decided to use Regression with ARIMA errors, as with this model we can use external factors and our dataset has three predictors namely Wind Speed, Theoretical Power Curve, and Wind Direction. Before running the algorithm, we want to see the relationships of these variables with Power production.

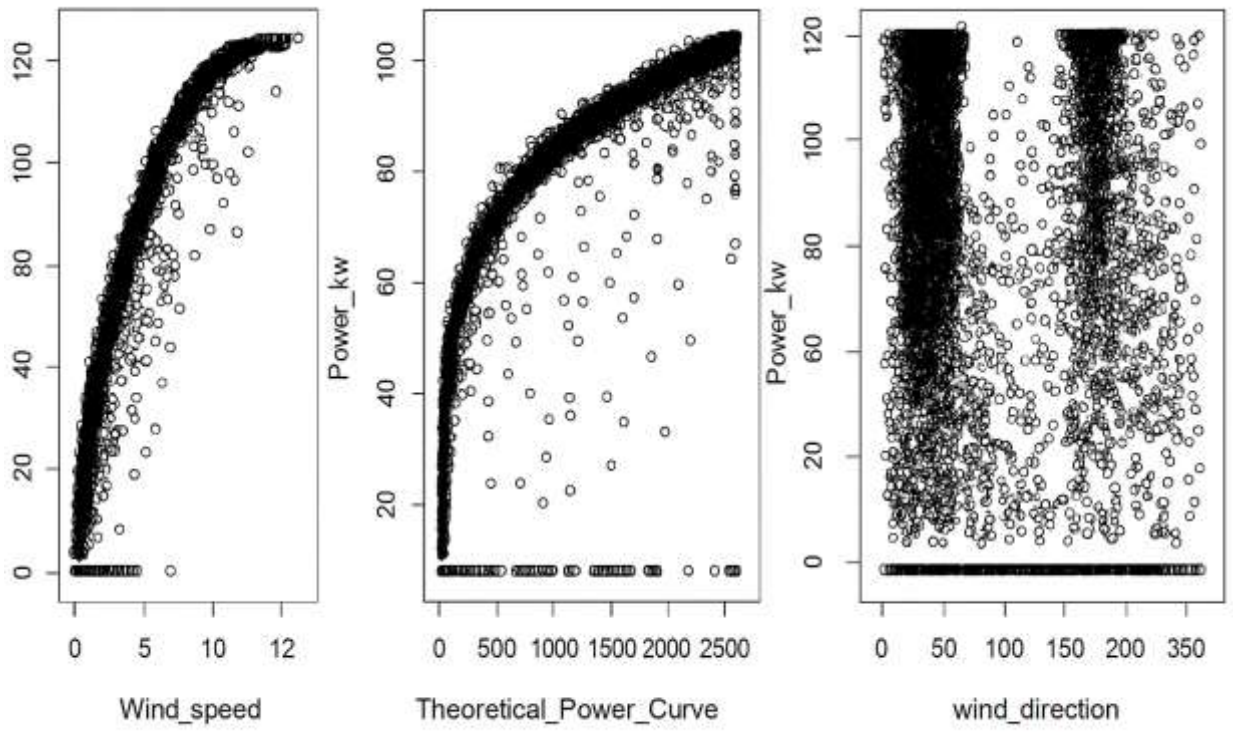


Figure 6: Scatter Plot Relationships Between Predictors and Power Output

From the graph we can see that wind speed and the theoretical power curve have a positive relationship with power output. Graphically wind direction does not have a strong correlation with power production, but we will keep it. Wind direction, despite lacking a clear graphical correlation, could still influence power output in complex or non-linear ways that may be captured in a comprehensive statistical model.

The model used here is an ARIMA(0,1,2) for the residuals of the regression model. This means that no autoregressive terms were used (0), the series was different once to make it stationary (1), and it has two moving average terms (2). The negative values for the moving average coefficients indicate that errors in prediction are likely to reverse sign in future periods.

The fact that wind direction has a negative coefficient suggests that an increase in wind direction (presumably measured in degrees from North) will decrease the power output slightly, holding everything else constant, even though this relationship was not initially obvious in the graphical analysis. This underlines the complexity of the relationship between wind direction and power output. From the accuracy measurement, we can see that the ARIMAX model has improved accuracy significantly with a test of RMSE of 20.8.

4.5 Dynamic harmonic regression

Dynamic regression with Fourrier terms can handle data that has very long seasonality. Our dataset has both daily and weekly seasonality, which are pretty long to handle by time series forecasting models such as ETS or ARIMA. So we have run Dynamic regression with both daily and weekly seasonality. While running code for this model, we wanted to select the best K for daily and weekly seasonality based on RSME, but it was running hours and hours. We tried to apply the code in a small sample of our data set with parallel computing.

Therefore, we have decided to run one simple model, one model that is moderately complex, and a complex model. Three models that we run are:

Model	Type	ME	RMSE	MAE	MPE	MAPE	MASE	RMSSE	ACF1
K = 1 & 8	Test	-99.68247	104.595	99.68247	−Inf	Inf	NaN	NaN	-
K = 7 & 30	Test	-102.91791	107.7452	102.91791	−Inf	Inf	NaN	NaN	-
K = 10 & 53	Test	-99.2812	104.2728	99.2812	−Inf	Inf	NaN	NaN	-

Table 8: Forecast Accuracy of Dynamic Harmonic Regression Models with Varying Fourier Terms (K)

When we have given a look at the RMSE test, it appears that a simple model is almost as good as the most complex model.

5 Data Analysis and Modelling with Predictive Methods

5.1 Linear Regression

Although linear regression is one of the simplest machine learning tools, it is one of the powerful tools to use. As a predictive model, we have used it. We have run two regression models. In the first model, we have used trend, and seasonality as predictors and, in the second model, we have used wind speed, theoretical power curve, and wind direction.

Coefficient	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-6.2987515	2.3535447	-2.676	0.00746
trend()	0.0014662	2.3535447	8.038	<2e-16
season(day2)	62.5261024	2.3535447	19.936	<2e-16
season(day3)	62.2807311	2.3535447	19.858	<2e-16
season(day4)	62.6635714	2.3535447	19.98	<2e-16
season(day5)	62.390005	2.3535447	20.065	<2e-16
season(day6)	61.1824261	2.3535447	19.508	<2e-16
season(day7)	58.0412802	2.3535447	18.507	<2e-16
season(day8)	54.2250363	2.3535447	17.29	<2e-16

Table 9: Coefficient Estimates from Linear Regression Model with Trend and Daily Seasonality

If we give a look at the result, we can see that all model coefficients were found to be highly significant ($p < 0.001$). The model, however, explained only a modest portion of the variability in the data, with an adjusted R-squared of 0.09471. Despite the significant coefficients, further refinement might be needed to improve the model's explanatory power. Now we will see how this model is doing in terms of test accuracy. It appears that linear regression with trend and seasonality has improved the RMSE (56.5) a lot. Linear regression with external predictors has even improved the predictors further with the test RMSE of 29.9.

5.2 Boosted Trees

The gbm function from the gbm library in R was used to fit this model.m. The distribution of the target variable, 'Power_kw', was assumed to be Gaussian. A total of 5000 decision trees were grown, with a maximum depth of 4 levels in each tree. This was achieved by setting n.trees = 5000 and interaction.depth = 4.

The `summary` function for a gbm model provides a measure of the relative influence (rel.inf) of each predictor variable, which is an indication of how important that variable is for the model's predictions.

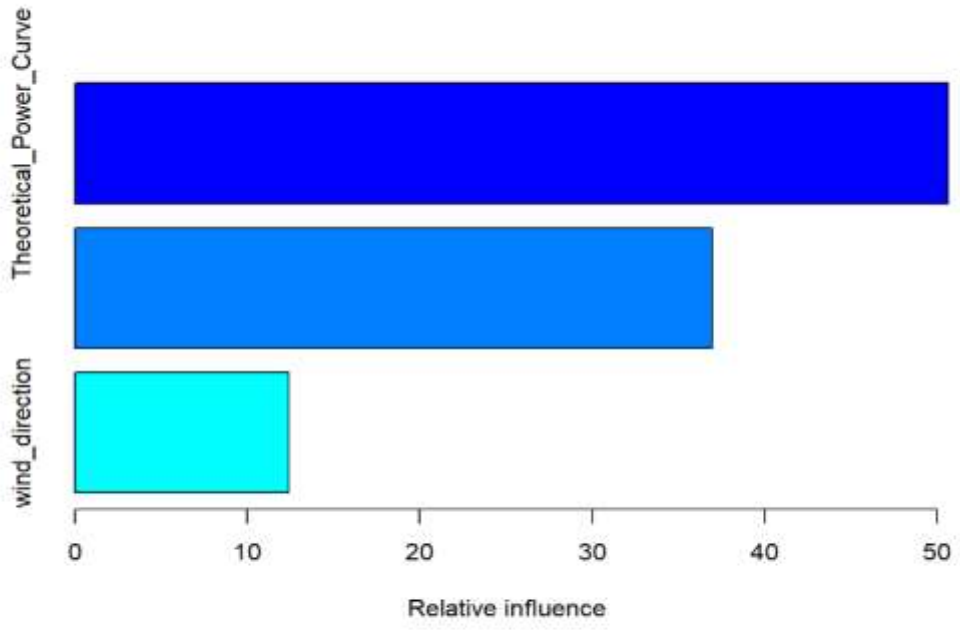
In our case:

`Theoretical_Power_Curve` is the most important predictor with a relative influence of 50.63%, indicating that this variable contributes the most to the model's predictive capability. `Wind_speed` is the second most influential predictor, contributing 36.97% to the prediction. `wind_direction` has the least relative influence among the three variables with 12.40%.

It's worth noting that these percentages are normalized to sum to 100% across all the predictors, and a higher percentage indicates a stronger relationship with the response variable in the context of the model.

This suggests that, according to the model, the amount of power produced by the turbine is most strongly associated with the Theoretical Power Curve, followed by the Wind Speed, and then the Wind Direction.

```
summary(boost_turbine)
```




```
##                                         var  rel.inf
## Theoretical_Power_Curve Theoretical_Power_Curve 50.62952
## Wind_speed                       Wind_speed 36.97149
## wind_direction                 wind_direction 12.39898
```

Figure 7: Variable Importance from Boosted Trees Model Predicting Power Output

From the partial dependence plot, we can see the same effect.

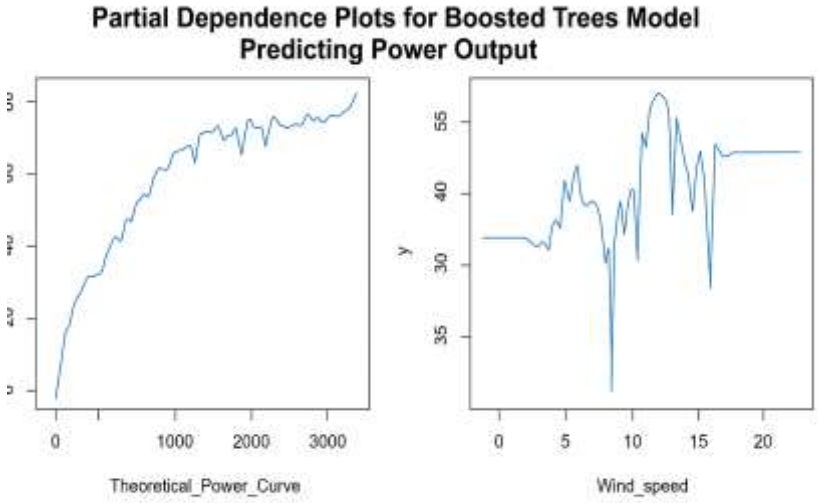


Figure 8: In terms of test RMSE, this model is doing far better than any model so far we have used with test RMSE of 16.051.

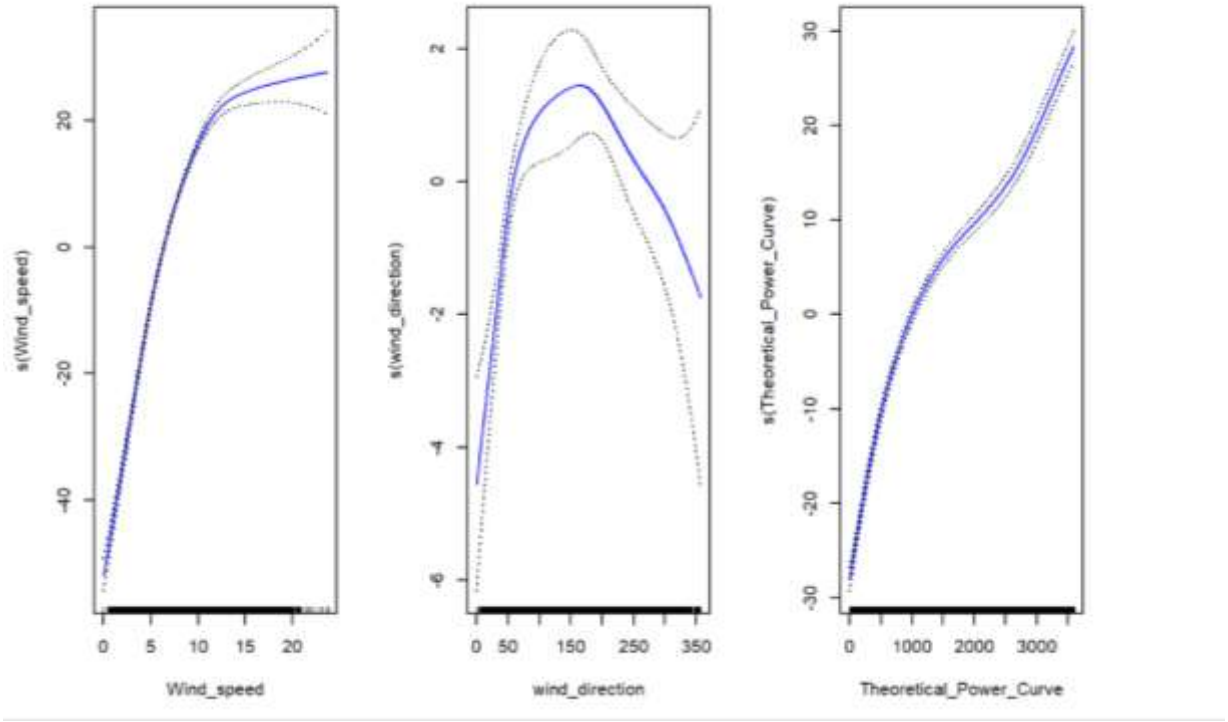
5.3 Generalized Additive Models (GAM)

We have fitted and compared three Generalized Additive Models(GAM). gam_fit1 uses smooth functions of all predictor variables with automatic selection of the degrees of freedom for each smooth term. gam_fit2 also applies smooth functions but sets specific degrees of freedom for each predictor. gam_fit3 applies a linear function to 'Wind_speed' and 'wind_direction' while keeping 'Theoretical_Power_Curve' as a smooth function. The Analysis of Deviance table produced by the anova() function compares these models.

From the ANOVA results, the Model 3 shows a highly significant improvement over both Model 1 and Model 2 (p-value < 2.2e-16), indicating that this model fits the data significantly better. However, the negative degrees of freedom and larger residual deviance in Model 3 suggest complexity and potentially overfitting. Given this, while the ANOVA suggests Model 3 as the best fit statistically, we should consider potential overfitting and interpret the model with caution. It's also important to remember that the "best" model is not only about statistical significance but also depends on the context.

For visualizing the estimated smooth functions for each predictor in the gam_fit1 model we have used plot.Gam() from the mgcv package. These plots help us to understand the shape of the relationship between each predictor and the response. The se=TRUE option means that the standard error bands will be included in the plots.

Figure 9: Estimated Smooth Functions from GAM Model (gam_fit1) for Wind Speed, Wind Direction, and Theoretical Power Curve



Now if we focus on test RMSE, it appears that model one is better than third model with test RMSE of 29.66 and 30.00 respectively. As we have already mentioned, choosing model 3 based on the training dataset may produce a result of overfitting.

6. Comparing Test RMSE

Models	RMSE
Boosted Tree	16.05
Regression with ARIMA errors (ARIMAX)	20.8

Generalized Additive Models (GAM)	29.7
Linear regression with external factors	29.9
Linear regression with trend, and seasonality	56.5
SNAIVE	99
ETS (automatically generated)	102
Dynamic harmonic regression (best model)	104.27
ARIMA	152

Table 10: Model Accuracy (RMSE)

In our study, we compared the predictive performance of eight different models by using the Root Mean Square Error (RMSE) as a measure of model accuracy. According to our results, the Boosted Tree model provided the most accurate forecasts with the lowest RMSE of 16.05, followed closely by the ARIMAX model with an RMSE of 20.8. The Generalized Additive Models (GAM) and the Linear regression with external factors showed comparable performance, with RMSE values of 29.7 and 29.9, respectively. The Linear regression with trend and seasonality model demonstrated a moderate predictive accuracy with an RMSE of 56.5. On the other hand, SNAIVE, ETS (automatically generated), Dynamic harmonic regression, and ARIMA models showed significantly less accuracy in their predictions, with RMSE values of 99, 102, 104.27, and 152, respectively. These results suggest that machine learning techniques such as Boosted Trees and time series models like ARIMAX can provide more precise forecasts in our case study compared to other considered models.

7. Conclusion

This study evaluated a range of statistical and machine learning models for short-term wind power forecasting using SCADA data from a commercial wind turbine. Among the nine models assessed, the Boosted Tree model outperformed all others with the lowest RMSE of 16.05, followed by the ARIMAX model with an RMSE of 20.8. Generalized Additive Models and linear regression using meteorological inputs also yielded reasonably strong predictive performance.

Our results underscore the importance of incorporating exogenous variables such as wind speed, wind direction, and theoretical power curves to improve forecast accuracy. Moreover, the superior performance of ensemble learning techniques—particularly Gradient Boosted Trees—demonstrates the value of machine learning approaches in handling non-linear, high-frequency renewable energy data.

From a practical standpoint, the findings support the use of ML-enhanced forecasting systems in operational grid environments to improve scheduling, load balancing, and integration of intermittent renewable resources. The comparison also reveals that while simpler models like SNAIVE or ETS offer interpretability, they are significantly less effective in capturing the complexities of wind power generation.

Future research may explore hybrid deep learning frameworks, real-time adaptive modeling, or spatiotemporal forecasting by incorporating wind farm networks and meteorological forecasts. Additionally, extending the model evaluation to larger datasets and multi-turbine scenarios could enhance generalizability and operational relevance.

Statements and Declarations
Funding: Please add: This research received no external funding
Conflicts of Interest: The authors declare no conflict of interest
Acknowledgments: Not Applicable

References

[1] International Energy Agency (IEA-2023) https://www.iea.org/reports/renewables-2023
[2] Zhang, J., Wang, J., & Wang, X. (2014). Review on probabilistic forecasting of wind power generation. Renewable and Sustainable Energy Reviews, 32, 255–270. https://doi.org/10.1016/j.rser.2014.01.033
[3] Foley, A. M., Leahy, P. G., Marvuglia, A., & McKeogh, E. J. (2012). Current methods and advances in forecasting of wind power generation. Renewable Energy, 37(1), 1–8. https://doi.org/10.1016/j.renene.2011.05.033
[4] Giebel, G., et al. (2011). The State-of-the-Art in Short-Term Prediction of Wind Power: A Literature Overview. ANEMOS.plus, EU Project.
[5] Soman, S. S., Zareipour, H., Malik, O., & Mandal, P. (2010). A review of wind power and wind speed forecasting methods with different time horizons. North American Power Symposium (NAPS).

[6] Zhou, H., Hu, J., & Zhang, Z. (2011). Short-term wind power forecasting based on Markov chain and neural network. Energy Conversion and Management, 52(2), 1244–1251. https://doi.org/10.1016/j.enconman.2010.09.025
[7] Rashid, M. M., et al. (2020). Wind power prediction using random forest machine learning technique. Energy Reports, 6, 2210–2220. https://doi.org/10.1016/j.egyr.2020.07.033
[8] Singh, A., & Kumar, A. (2021). Forecasting wind power using hybrid machine learning models. Renewable and Sustainable Energy Reviews, 135, 110234. https://doi.org/10.1016/j.rser.2020.110234
[9] Wang, J., et al. (2017). Short-term wind speed forecasting based on hybrid model. Energy Conversion and Management, 136, 443–451.
[10] Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). Time Series Analysis: Forecasting and Control (5th ed.). Wiley.
[11] Hyndman, R. J., & Athanasopoulos, G. (2021). Forecasting: Principles and Practice (3rd ed.). OTexts. https://otexts.com/fpp3/
[12] Hyndman, R. J., Koehler, A. B., Ord, J. K., & Snyder, R. D. (2008). Forecasting with Exponential Smoothing: The State Space Approach. Springer.
[13] Hastie, T., & Tibshirani, R. (1990). Generalized Additive Models. Chapman and Hall/CRC.
[14] Xie, L., et al. (2014). Wind power forecasting and its applications: A literature review. Energy Reports, 1, 1–6.
[15] Natekin, A., & Knoll, A. (2013). Gradient boosting machines: A tutorial. Frontiers in Neurorobotics, 7, 21. https://doi.org/10.3389/fnins.2013.00021
[16] Gao, J., Wang, X., & Zhang, L. (2019). Hybrid modeling approach based on ARIMA and SVM for wind speed forecasting. Energy Reports, 5, 1042–1049. https://doi.org/10.1016/j.egyr.2019.07.011
[17] Box, G. E., & Cox, D. R. (1964). An analysis of transformations. Journal of the Royal Statistical Society: Series B, 26(2), 211–252.
[18] Taylor, J. W., & Hyndman, R. J. (2017). A seasonal state space model for exponential smoothing. International Journal of Forecasting, 34(2), 239–254.
[19] Ridgeway, G. (2006). Generalized Boosted Models: A guide to the gbm package. CRAN.
[20] Wood, S. N. (2017). Generalized Additive Models: An Introduction with R (2nd ed.). CRC Press.