International Journal of Management and Data Analytics (IJMADA)

Int. J. Management and Data Analytics, Vol. 5 (1), 89-97
ISSN: 2816-9395
Journal Homepage: http://ijmada.com
DOI: https://doi.org/10.5281/zenodo.14948203

Open-Source Climate Modeling for Solar and Wind Energy Estimation: Validation with SAM Data and Experimental Measurements

O. S. Abd El-Kawi[1, 2]

[1]*Mechanical Engineering Dept., Faculty of Engineering, Al-Baha University, Saudi Arabia*
[2]*Reactors Department, Nuclear Research Center, Egyptian Atomic Energy Authority, Egypt*

Received: Jan, 2025, Published: Mar, 2025

ARTICLE INFO
Keywords:
Climate modeling; Climatic variables; Solar irradiance; Solar energy models

Cite: Abd El-Kawi, O. S. (2025). Open-Source Climate Modeling for Solar and Wind Energy Estimation: Validation with SAM Data and Experimental Measurements. International Journal of Management and Data Analytics, 5(1), 89-97. https://doi.org/10.5281/zenodo.14948203

ABSTRACT

Climate modeling for solar and wind energy estimation involves predicting weather parameters like sunlight, wind speed, and atmospheric conditions to assess energy potential. The current study introduces reliable predictions which helps to optimize the placement and efficiency of solar panels and wind turbines. Continuous refinement of models enhances renewable energy planning and integration into power grids. This study assesses the performance of an open-source model in predicting environmental parameters, with a particular focus on bias as the primary evaluation metric. The predictions generated by the current code output are compared with observed data and results from the System Advisor Model (SAM), which utilizes Typical Meteorological Year (TMY) data for the Al-Baha region. The findings reveal that the current code output generally exhibits lower biases across most parameters compared to SAM, underscoring its reliability and adaptability. For temperature, the current code output shows a bias of -4.62 compared to SAM's -2.01, with both models underestimating. In the case of relative humidity, the current code output demonstrates a bias of 7.59, slightly higher than SAM's 7.14, indicating comparable alignment with actual data. Regarding pressure, the current code output exhibits a bias of 8.51, which is somewhat larger than SAM's 2.79, though both remain within acceptable accuracy. For wind speed, the bias of the current code output is 0.82, higher than SAM's 0.47, with both models slightly overestimating. In contrast, for wind direction, the current code output achieves a substantially smaller bias of 17.35 compared to SAM's 37.78. For direct normal irradiance (DNI), the Current Code Output has a bias of -9.11, reflecting a minor underestimation, whereas SAM significantly overestimates with a bias of 44.85. The open-source nature of the current code presents considerable potential for further refinement. Its customizable and adaptable framework allows for recalibration to specific datasets, fostering continuous improvements in accuracy. By addressing biases in temperature and pressure, the model could evolve into a highly reliable and precise tool for environmental parameter prediction. Its flexibility supports collaborative development, making it a robust and viable alternative to SAM's TMY-based data for modeling in the Al-Baha region.

1. INTRODUCTION

The rising global demand for renewable energy has driven the need for accurate tools to assess the potential of solar and wind energy. These renewable energy sources are critical for achieving energy sustainability and reducing the negative impacts of fossil fuel consumption on climate change [1, 2]. Solar and wind energy production is influenced by climatic variables such as solar irradiance, temperature, atmospheric pressure, and wind speed, all of which exhibit significant spatial and temporal variability [3, 4]. Accurate modeling of these factors is essential for optimizing energy systems, planning infrastructure, and ensuring the economic feasibility of renewable energy projects [5, 6]. Computational modeling has emerged as a vital approach for estimating renewable energy potential. Solar energy models typically incorporate parameters such as atmospheric transmittance, cloud cover, and solar zenith angle, while wind energy models are driven by wind velocity profiles and air density variations [7, 8]. These models provide detailed insights into resource availability and system performance under specific climatic conditions, making them indispensable tools for designing and managing renewable energy projects [9, 10]. Recent advancements in research have focused on integrating meteorological datasets with computational tools to improve the accuracy of energy estimation models [11, 12]. However, ensuring model accuracy and reliability remains a major challenge, particularly when applying these models across diverse geographic and climatic regions [13, 14]. This study develops a Python-based computational framework to estimate solar and wind energy potential using climatic factors. Python's extensive libraries and robust computational capabilities make it an ideal platform for implementing flexible and scalable models [15, 16]. The primary objectives of this research are to (i) design and

implement a computational model that integrates climate factors for solar and wind energy estimation and (ii) validate the model using real-world experimental data. This study contributes to the growing body of knowledge on renewable energy resource assessment by offering a versatile tool for researchers and practitioners involved in energy planning and development. By addressing gaps in model validation and improving regional applicability, the research provides valuable insights for optimizing renewable energy systems. The findings aim to benefit policymakers, developers, and scientists committed to advancing renewable energy adoption. Furthermore, the broader implications include enhanced decision-making processes and a deeper understanding of the relationship between climate factors and renewable energy potential.

2. Computational Model Analysis

This section provides a detailed overview of the computational framework developed to estimate solar and wind energy potential. It addresses key processes such as the calculation of Direct Normal Irradiance (DNI), the modeling of wind speed and direction, the evaluation of pressure gradients, and the integration of atmospheric parameters. Additionally, the section outlines the Python-based algorithms utilized for these computations.

2.1 Direct Normal Irradiance (DNI) Calculation

DNI represents the amount of solar radiation received per unit area on a surface-oriented perpendicular to the Sun's rays. It can be calculated as follows [17]:

$$\text{DNI} = G_{\text{sc}} \times E_0 \times \exp(-\tau_{ext} \times m) \quad (1)$$

Where:

G_{sc} = Solar constant (updated value 1361 W/m²) [18]

E_0 = Eccentricity correction factor

τ_{ext} = Atmospheric extinction optical depth

m = Air mass (relative optical path length)

2.1.1 Eccentricity Correction Factor (E_0) Calculation [19]:

$$E_0 = 1 + 0.033 \cdot \cos\left(\frac{360 \cdot \text{n}}{365}\right) \quad (2)$$

Where:

n = Day of the year (1 to 365)

2.1.2 Air Mass (m) Calculation:

For zenith angles θ_z less than 90°:

$$m = \frac{1}{\cos(\theta_z)} \quad (3)$$

For larger zenith angles, the Kasten and Young (1989) approximation is used[4]:

$$m = \frac{1}{\cos(\theta_z)+0.50572(96.07995°-\theta_z)^{-1.6364}} \quad (4)$$

Where:

θ_z = Solar zenith angle in degrees

2.1.3 Solar Zenith Angle (θ_z) Calculation [20]:

$$\cos(\theta_z) = \sin(\phi)\sin(\delta) + \cos(\phi)\cos(\delta)\cos(h) \quad (5)$$

Where:

ϕ = Latitude (in radians)

δ = Solar declination angle (in radians)

h = Hour angle (in radians)

2.1.4 Solar Declination Angle (δ) Calculation[21]:

$$\delta = 23.45°\sin\left(\frac{360°}{365}(284+n)\right) \quad (6)$$

Convert δ to radians when used in calculations.

2.1.5 Hour Angle (h) Calculation[21]:

$$h = 15°(\text{Local Solar Time} - 12) \quad (7)$$

2.1.6 Atmospheric Extinction Coefficient (τ_{ext}) Calculation[17, 22]:

$$\tau_{\text{ext}} = \tau_{\text{rayleigh}} + \tau_{\text{aerosol}} + \tau_{\text{ozone}} + \tau_{\text{gases}} + \tau_{\text{water vapor}} \quad (8)$$

Each component is calculated as described in the following sections.

2.1.6.1 Rayleigh Scattering (τ_{rayleigh}) [22]:

$$\tau_{\text{rayleigh}} = \frac{P}{P_0} \times 0.008735\lambda^{-4} \quad (9)$$

Where:

P = Local atmospheric pressure (Pa)

P_0 = Standard atmospheric pressure (101325 Pa)

λ = Wavelength (in micrometers)

2.1.6.2 Aerosol Optical Depth (τ_{aerosol})[22, 23]:

$$\tau_{\text{aerosol}} = \beta\lambda^{-\alpha} \quad (10)$$

Where:

β = Aerosol turbidity coefficient

α = Ångström exponent

2.1.6.3 Ozone Absorption (τ_{ozone}) [3]:

$$\tau_{\text{ozone}} = k_{\text{ozone}}u_{\text{ozone}} \quad (11)$$

Where:

k_{ozone} = Ozone absorption coefficient

u_{ozone} = Total ozone column amount (atm·cm)

2.1.6.4 Water Vapor Absorption ($\tau_{\text{water vapor}}$)[17, 22]:

$$\tau_{\text{water vapor}} = k_{\text{water}} u_{\text{water}} \quad (12)$$

Where:

k_{water} = Water vapor absorption coefficient

u_{water} = Precipitable water vapor (cm)

2.1.6.5 Gaseous Absorption (τ_{gases})[22]:

$$\tau_{\text{gases}} = k_{\text{gases}} u_{\text{gases}} \quad (13)$$

Where:

k_{gases} = The gaseous absorption coefficient

u_{gases} = The gaseous column amount (or optical path amount) along the line of sight.

2.2 Wind Speed and Direction

Wind speed and direction are modeled with seasonal and diurnal variations[24-26].

2.2.1 Wind Speed

$$\text{Wind Speed} = \text{A1} + 2 \cdot \sin\left(\frac{2\pi \cdot (\text{day\_of\_year} - 80)}{365}\right) + 1 \cdot \sin\left(\frac{2\pi \cdot (\text{hour} - 6)}{24}\right) \quad (14)$$

A1: Base wind speed.

2.2.2 Wind Direction

$$\text{Wind direction} = \left(\text{B} + 30 \cdot \sin\left(\frac{2\pi \cdot (\text{day of year} - 80)}{365}\right) + 15 \cdot \sin\left(\frac{2\pi \cdot (\text{hour} - 12)}{24}\right)\right) \mod 360 \quad (15)$$

B: Base wind direction[27].

2.3 Atmospheric Parameters

In this section atmospheric parameters calculation are discussed in details.

2.3.1 Temperature Calculation[27, 28]

The temperature is modeled as a function of the day of the year and altitude.

$$T_k = T_{\text{avg}} + A.\sin\left(\frac{2\pi \cdot (\text{day of year} - \text{phase shift})}{365}\right) \quad (16)$$

$$T = T_{\text{base}} + \text{lapse rate} \cdot \text{height} \quad (17)$$

Where:

T_{avg}: Average annual temperature at sea level (30°C).

A: Amplitude of temperature variation (10°C).

phase shift: Phase shift for mid-June peak (173 days).

lapse rate: Temperature lapse rate (-0.0065°C/m).

Height: Altitude in meters.

2.3.2 Relative Humidity Calculation [29, 30]

Relative humidity is calculated using the Hyland-Wexler model for saturation vapor pressure.

$$\text{Saturation vapor pressure} = \exp\left(54.842763 - \frac{6763.22}{T_k} - 4.210 \cdot \ln(T_k) + 0.000367 \cdot T_k + \tanh\left(0.0415 \cdot (T_k - 218.8)\right) \cdot \left(53.878 - \frac{1331.22}{T_k} - 9.44523 \cdot \ln(T_k) + 0.014025 \cdot T_k\right)\right) \quad (18)$$

$$\text{Relative humidity} = \left(\frac{\text{actual vapor pressure}}{\text{saturation vapor pressure}}\right) \cdot 100 \quad (19)$$

- T_k: Temperature in Kelvin.
- Actual vapor pressure: Saturation vapor pressure at the dew point temperature.

2.3.3 Pressure Calculation[25, 27]

Atmospheric pressure is calculated using the barometric formula.

$$P = P_0 \cdot \exp\left(-\frac{g \cdot M \cdot \text{height}}{R \cdot T_k}\right) \quad (20)$$

- P_0: Sea-level pressure (101325 Pa).
- g: Gravitational acceleration (9.80665 m/s²).
- M: Molar mass of air (0.0289644 kg/mol).
- R: Universal gas constant (8.31432 N·m/(mol·K)).
- T_k: Temperature in Kelvin.

3. Overview of the Developed Software

This paper presents a Python-based software tool designed to simulate and visualize weather data based on user inputs. The software utilizes mathematical models to calculate key weather parameters, including solar radiation (Direct Normal Irradiance, DNI), temperature, humidity, pressure, wind speed, and wind direction. These parameters can be analyzed over different time intervals, such as hourly, daily, or monthly, depending on user requirements. A graphical user interface (GUI) is integrated into the software to enhance usability and streamline data entry. Through the interface, users can input essential parameters such as latitude, longitude, altitude, start date, and the forecasting period. The interface also provides real-time visualization of the generated weather forecasts, allowing users to assess trends and variations instantly. To ensure efficient data handling, the software automatically saves forecasted data in an Excel file, enabling further analysis and external processing. Additionally, the tool supports graphical representations of the results, making it easier to interpret complex datasets. The visualization features help users identify patterns in solar and wind conditions, which are crucial for energy estimation and planning. Fig. 1 illustrates the software's workflow through a flowchart,

detailing each step in the data processing and visualization sequence. Fig. 2 presents a comprehensive screenshot of the Python code in action, showing the structure of the input parameters and a sample of the corresponding output results. This figure demonstrates how the input data is processed, transformed into meaningful predictions, and displayed within the interface, providing a clear overview of the model's capabilities and performance.

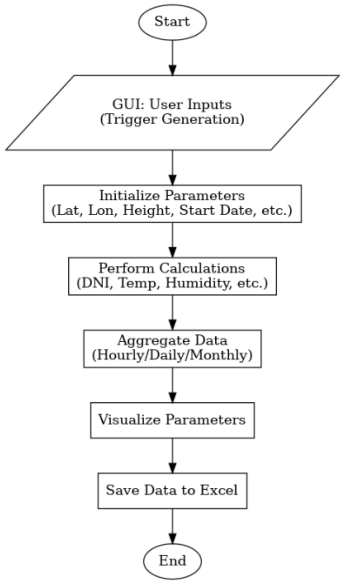
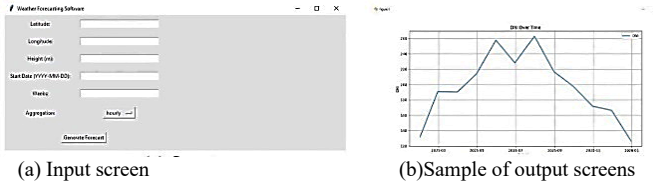


Figure 1. The flowchart diagram of the current Python code.



(a) Input screen (b)Sample of output screens

Figure 2. Screenshot of the current Python code inputs and outputs

4. Experimental and Theoretical Comparison with Error Analysis

Measurements obtained from the Al-Baha region, located in the southwestern part of Saudi Arabia at a latitude of 20.5°N and a longitude of 41.7°E [31]. The experimental data collected from this region serves as a benchmark for evaluating the accuracy of the theoretical models. Additionally, the comparison is extended to include data derived from the System Advisor Model (SAM)[32], which relies on the typical meteorological year file for the Al-Baha region sourced from [33].To evaluate the error analysis. Bias concept [34-37]. Bias in error analysis measures the average difference between the predicted values and the actual (true) values. It indicates whether a model systematically overpredicts or underpredicts the actual values over time. The following Formula is used for Bias calculations:

$$\text{Bias} = \frac{1}{m}\sum_{i=1}^{m}(\hat{y}_i - y_i) \quad (21)$$

Where:

$\hat{y}_i$ = Predicted value

y_i = Actual value

m = Number of observations

Positive Bias Indicates that the model, on average, overestimates the actual values. Where negative Bias Indicates that the model, on average, underestimates the actual values. Finally, zero Bias Suggests that the model's predictions are, on average, neither overestimating nor underestimating the actual values. This does not mean the predictions are perfectly accurate; errors could still exist but might cancel each other out. Bias used to identify systematic errors, model calibration, and fairness and reliability.

5. Results and Discussions

After the text edit has been completed, the paper is ready for the template. Fig. 3 provides a comparative analysis of Direct Normal Irradiance (DNI) values derived from actual measurements, outputs of the current code, and SAM data. The SAM data (represented by the blue line) consistently overestimates DNI, while the current code output (depicted by the green line) shows a much closer alignment with actual readings (red line). The current code effectively captures seasonal variations and trends. While minor discrepancies are observed during specific months, such as July and October, the overall performance of the current code demonstrates superior accuracy and lower error metrics compared to SAM data. This underscores the reliability of the current code. Additional refinements and improved visualizations could further enhance its performance and usability.

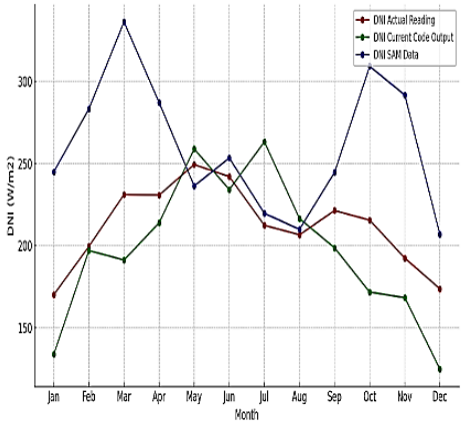
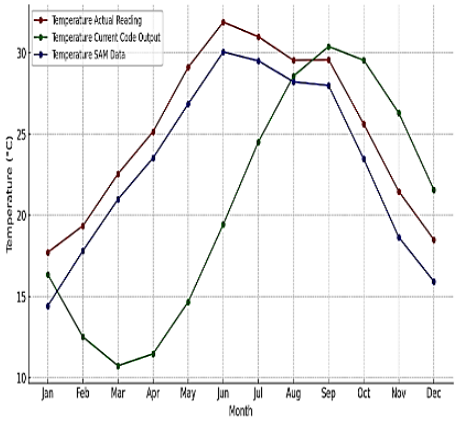


Figure 3. DNI comparison between current code, SAM data, and actual measurements.

Fig. 4 illustrates the comparison of temperature data between actual measurements, outputs from the current code, and SAM data. The current code output (green line)

displays significant deviations from the actual readings (red line) during the initial months of the year, with notable underestimations. From May onward, the current code begins to overestimate temperatures, with these discrepancies becoming more pronounced in the latter months. SAM data (blue line), on the other hand, exhibits better alignment with actual readings during the first half of the year but tends to underestimate temperatures in the second half. Overall, the current code output shows weaker performance in predicting temperatures compared to SAM data, with larger deviations from actual values. While SAM data demonstrates relative accuracy in the earlier months, its accuracy diminishes in the later months, emphasizing the need for recalibration in both models to achieve greater consistency throughout the year. To provide a clearer understanding of these differences, visual tools such as



residual plots could be used to better highlight the magnitude and distribution of errors.

Figure 4. Temperature comparison between current code, SAM data, and actual measurements.

Fig. 5 compares relative humidity data from actual readings, current code outputs, and SAM data. The SAM data (blue line) consistently overestimates relative humidity in the early months, particularly in January and February, and significantly deviates in November and December. In contrast, the current code output (green line) remains relatively stable throughout the year but consistently underestimates relative humidity compared to the actual readings (red line), especially in the first half of the year. In months like June and August, the current code output aligns better with actual readings, showing strength in capturing localized patterns. However, both the SAM data and the current code output fail to adequately track sudden variations in actual readings, such as the dip in June and the spike in August. Overall, while the current code output demonstrates stability, it underestimates relative humidity, and the SAM data, though dynamic, shows excessive deviations from actual readings. Improvements in capturing seasonal variability could enhance model accuracy.

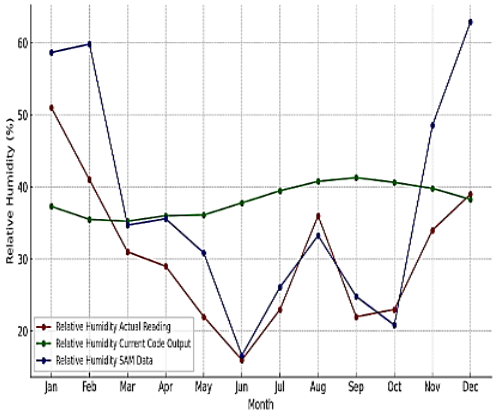


Figure 5. Comparison of code output for relative humidity values with SAM data and actual measurements

Fig. 6 presents a comparison of pressure data (in millibars) between actual measurements, outputs from the current code, and SAM data. The current code output (green line) consistently overestimates pressure throughout the year, with the divergence from actual readings (red line) becoming particularly pronounced during the second half of the year. In contrast, SAM data (blue line) demonstrates closer alignment with actual readings during the first half of the year. However, it underestimates pressure values from June to August. The actual readings reveal a distinct seasonal pattern, with a noticeable dip in summer (June–July) and a gradual rise toward the end of the year. While SAM data captures this seasonal trend to some extent, its accuracy remains limited. The current code output, on the other hand, fails to reflect this seasonal variation and instead shows consistent overestimation, which undermines its reliability. Improving the current code to better account for seasonal fluctuations in pressure would significantly enhance its accuracy and overall performance. Such refinements could enable it to match or exceed the precision of SAM data in capturing seasonal trends.

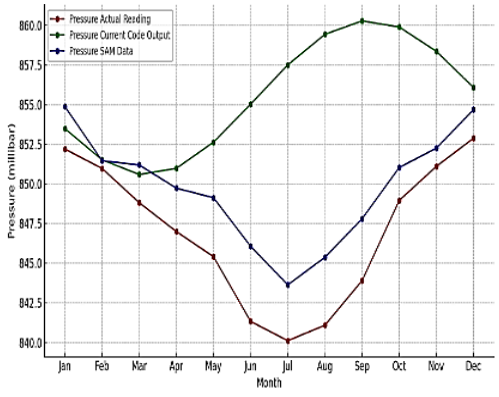


Figure 6. Comparison of code output for pressure values with SAM data and actual measurements

Fig. 7 compares wind speed data (in m/s) obtained from actual measurements, the current code outputs, and SAM data. The current code output (green line) consistently overestimates wind speed throughout the year, with particularly high deviations during the summer months (May to September). SAM data (blue line), on the other hand, shows less pronounced overestimation and aligns more closely with actual readings (red line), particularly during January to April and September to December. The actual readings display notable variability, including a sharp peak in July and dips in June and December. While SAM data captures these seasonal trends more effectively, the current code output exaggerates fluctuations, as evident in its pronounced peaks during June and October. Both models, however, fail to accurately reproduce the sharp rise observed in July in the actual readings. Overall, SAM data demonstrates better performance in tracking wind speed trends compared to the current code output. Nevertheless, both models require further refinement to improve their accuracy, especially during months with significant variability in wind speed. Addressing these discrepancies would enhance the reliability of both models in wind speed prediction.

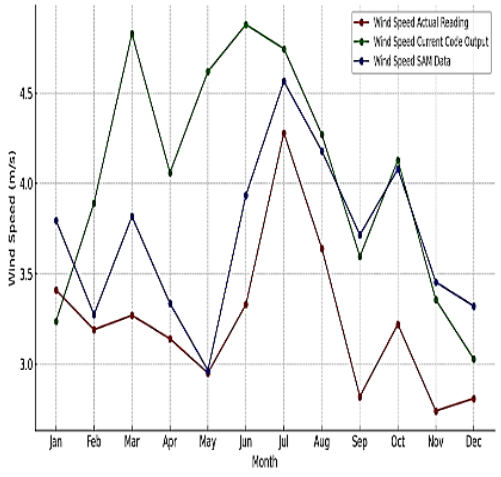


Figure 7. Comparison of code output for wind speed values with SAM data and actual measurements

Fig. 8 compares wind direction data (in degrees) from actual readings, current code outputs, and SAM data. The current code output (green line) shows a stable trend throughout the year but consistently overestimates wind direction compared to the actual readings (red line). The SAM data (blue line) demonstrates large variability and often overestimates wind direction, particularly from February to August, and deviates significantly from actual readings. Notably, the actual readings show a peak in July, which is captured more accurately by the SAM data but not by the current code output. However, the SAM data exhibits exaggerated fluctuations in the earlier months (e.g., February and April) and an overcorrection in October, where it underestimates wind direction. Overall, the current code output is more stable but fails to track the seasonal dynamics of wind direction, while SAM data captures variability but overestimates or underestimates significantly in several months. Refining both models could improve alignment with actual readings and seasonal variations.

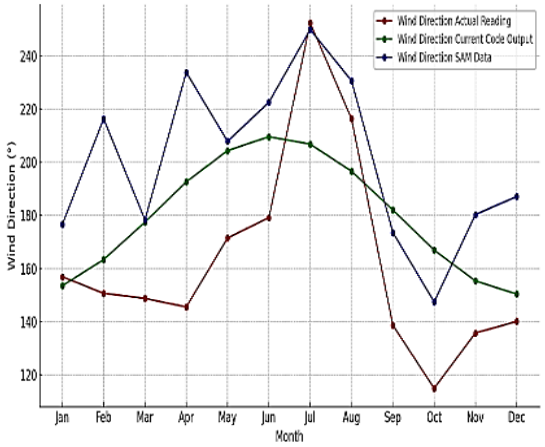


Figure 8. Comparison of code output for wind direction values with SAM data and actual measurements

The bar chart in Fig. 9 compares the bias values for six parameters; temperature, relative humidity, pressure, wind speed, wind direction, and DNI, between the current code output (green bars) and SAM Data (blue bars). For temperature, the current code output exhibits a bias of -4.62, while SAM Data has a bias of -2.01. Both models slightly underestimate temperature, with SAM Data showing marginally better performance. In relative humidity, the current code output has a bias of 7.59, whereas SAM Data shows a slightly smaller bias of 7.14. Both models overestimate humidity, though the SAM Data performs only slightly better in this case. For pressure, the current code output has a bias of 8.51 compared to SAM Data's bias of 2.79. Both models overestimate pressure, but SAM Data aligns more closely with actual values. In wind speed, the current code output demonstrates a bias of 0.82, while SAM Data shows a smaller bias of 0.47. Both models exhibit slight overestimation, with SAM Data performing better. When considering wind direction, the current code output has a bias of 17.35, significantly smaller than SAM Data's bias of 37.78. This indicates that SAM Data has substantial overestimation, while the Current Code Output provides more reliable results. For DNI (Direct Normal Irradiance), the current code output shows a bias of -9.11, indicating slight underestimation, whereas SAM Data exhibits a much higher bias of 44.85, reflecting severe overestimation. Overall, the current code output demonstrates superior performance across most parameters, particularly for DNI, wind direction, and relative humidity, where it achieves smaller biases compared to SAM Data. Although SAM Data outperforms the Current Code Output in pressure and temperature, its significant overestimation in other parameters highlights

the need for recalibration. The results depicted in the chart underscore the stability, adaptability, and overall reliability of the current code output in contrast to SAM Data.

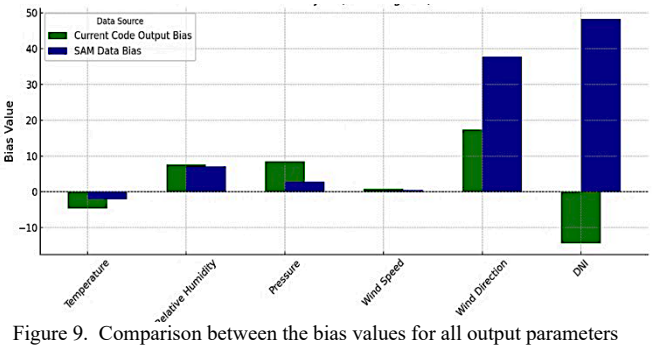


Figure 9. Comparison between the bias values for all output parameters for the current code with SAM data and actual measurements

Table 1: The bias values for all output parameters for current code and SAM data with actual measurements

Parameter	***Current Code Output Bias***	***SAM Data Bias***
Temperature	-4.61667	-2.00504
Relative Humidity	7.594307	7.137042
Pressure	8.505537	2.790267
Wind Speed	0.820238	0.469653
Wind Direction	17.34643	37.77975
DNI	-17.7914	35.50457

Finally, Fig. 10 compares the bias values, RMSE (Root Mean Square Error), and MAE (Mean Absolute Error) for two data sources: "Current Code Output Bias" and "SAM Data Bias." The bias values indicate the deviation of measurements from expected values across different parameters. The RMSE and MAE charts show that the SAM Data Bias has significantly higher error values compared to the Current Code Output Bias, particularly for parameters like Wind Direction and DNI. This suggests that the SAM Data has larger deviations and less consistency in comparison to the current code output.

6. Conclusions

The Current Code Output demonstrates promising performance across most parameters, particularly for DNI, Wind Direction, and Relative Humidity, where it consistently exhibits smaller biases compared to SAM Data. While it does show some areas of overestimation or underestimation, such as in Pressure and Temperature, these deviations are relatively minor compared to the significant overestimations observed in SAM Data. Its ability to closely align with actual readings in key parameters highlights its reliability and potential as a predictive tool.

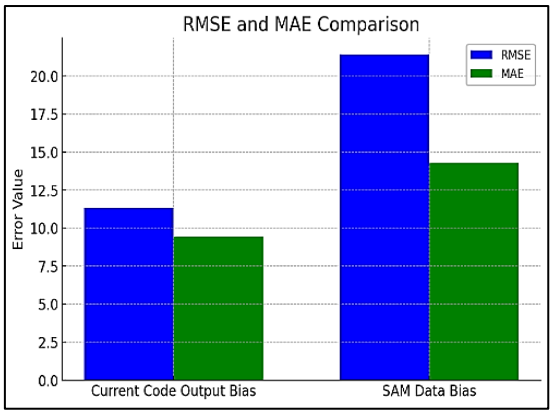


Figure 10. Comparison between the RMSE values and MAE values for the current code out bias with SAM data bias from actual measurements

One of the key strengths of the Current Code Output lies in its open-source nature**,** which allows for continuous improvement and customization. With further refinements, such as recalibration to reduce overestimation in Wind Speed and fine-tuning to capture seasonal patterns more effectively, the code can achieve even greater accuracy. The flexibility of open-source software enables collaboration and innovation, making it possible to adapt the code for specific use cases or datasets. In conclusion, while the Current Code Output already demonstrates a strong foundation, its open-source nature offers significant opportunities for improvement and optimization, ensuring it can become an even more robust and reliable tool for predictive modeling.

References

[1] Jacobson, M.Z. and M.A.J.E.p. Delucchi, Providing all global energy with wind, water, and solar power, Part I: Technologies, energy resources, quantities and areas of infrastructure, and materials. 2011. 39(3): p. 1154-1169.
[2] Abdullah-Al-Mahbub, M. and A.R.M.T.J.H. Islam, Current status of running renewable energy in Bangladesh and future prospect: A global comparison. 2023. 9(3).
[3] Bird, R.E. and R.L. Hulstrom, Simplified clear sky model for direct and diffuse insolation on horizontal surfaces. 1981, Solar Energy Research Inst.(SERI), Golden, CO (United States).
[4] Kasten, F. and A.T.J.A.o. Young, Revised optical air mass tables and approximation formula. 1989. 28(22): p. 4735-4738.
[5] Lara Fanego, V., Assessment and forecasting of solar resource: applications to the solar energy industry. 2017.
[6] da Guarda, E.L.A., et al., The influence of climate change on renewable energy systems designed to achieve zero energy buildings in the present: A case study in the Brazilian Savannah. 2020. 52: p. 101843.
[7] Tiwari, G.N. and S. Dubey, Fundamentals of photovoltaic modules and their applications. 2009: Royal Society of Chemistry.
[8] Burton, T., et al., Wind energy handbook. 2011: John Wiley & Sons.
[9] Kalogirou, S.A., Solar energy engineering: processes and systems. 2023: Elsevier.
[10] Liu, B.Y. and R.C.J.S.e. Jordan, The interrelationship and characteristic distribution of direct, diffuse and total solar radiation. 1960. 4(3): p. 1-19.
[11] Zhu, J., et al., Review and prospect of data-driven techniques for load forecasting in integrated energy systems. 2022. 321: p. 119269.

[12] Georgilakis, P.S. and N.D.J.I.T.o.p.s. Hatziargyriou, Optimal distributed generation placement in power distribution networks: models, methods, and future research. 2013. 28(3): p. 3420-3428.
[13] Van Rossum, G. and F. Drake, Python 3 Reference Manual CreateSpace Independent Publishing Platform. 2009.
[14] Oliphant, T.E.J.C.i.s. and engineering, Python for scientific computing. 2007. 9(3): p. 10-20.
[15] Atems, B. and C.J.E.P. Hotaling, The effect of renewable and nonrenewable electricity generation on economic growth. 2018. 112: p. 111-118.
[16] Quaschning, V., Understanding renewable energy systems. 2014: Routledge.
[17] Gueymard, C.A.J.S.E., Direct solar transmittance and irradiance predictions with broadband models. Part I: detailed theoretical performance assessment. 2003. 74(5): p. 355-379.
[18] Kopp, G. and J.L.J.G.R.L. Lean, A new, lower value of total solar irradiance: Evidence and climate significance. 2011. 38(1).
[19] Duffie, J.A., W.A. Beckman, and N. Blair, Solar engineering of thermal processes, photovoltaics and wind. 2020: John Wiley & Sons.
[20] Handoyo, E.A. and D.J.E.P. Ichsani, The optimal tilt angle of a solar collector. 2013. 32: p. 166-175.
[21] Duffie, J.A. and W.A. Beckman, Solar engineering of thermal processes. 1980: Wiley New York.
[22] Liou, H.-C.J.B.r., Regulation of the immune system by NF-κB and IκB. 2002. 35(6): p. 537-546.
[23] Ångström, A.J.T., Techniques of Determinig the Turbidity of the Atmosphere. 1961. 13(2): p. 214-223.
[24] Stull, R.B., An introduction to boundary layer meteorology. Vol. 13. 2012: Springer Science & Business Media.
[25] Holton, J.J.I.g.s., An introduction to dynamic meteorology. 1992. 48: p. 1-497.
[26] Oke, T.R., Boundary layer climates. 2002: Routledge.
[27] Wallace, J.M. and P.V. Hobbs, Atmospheric science: an introductory survey. Vol. 92. 2006: Elsevier.
[28] Hartmann, D.L., Global physical climatology. Vol. 103. 2015: Newnes.
[29] Hyland, R.W. and A. Wexler, FORMULATIONS FOR THE THERMODYNAMIC PROPERTIES OF THE SATURATED PHASES OF H2O FROM 173.15 TO 473.15 K. 1983.
[30] Bolton, D.J.M.w.r., The computation of equivalent potential temperature. 1980. 108(7): p. 1046-1053.
[31] AlGhamdi, S.A., Analysis of the Economic Feasibility of Renewable Energy (Wind and Solar Energies) in Al-Baha Region. 2024, Al-Baha University: Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia.
[32] Blair, N., et al., System advisor model (SAM) general description (Version 2017.9. 5). 2018, National Renewable Energy Laboratory (NREL), Golden, CO (United States).
[33] PVGIS, E.J.l.A.h.r.j.e.e.e.p.t.e.P., Photovoltaic geographical information system. 2018.
[34] Willmott, C.J. and K.J.C.r. Matsuura, Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. 2005. 30(1): p. 79-82.
[35] Janić, Z.I., Nonsingular implementation of the Mellor-Yamada level 2.5 scheme in the NCEP Meso model. 2001.
[36] McKeen, S., et al., Assessment of an ensemble of seven real-time ozone forecasts over eastern North America during the summer of 2004. 2005. 110(D21).
[37] Kumar, Y., et al., Wind energy: Trends and enabling technologies. 2016. 53: p. 209-224.

APPENDIX: NOMENCLATURE AND ABBREVIATIONS

Symbol	Definition
G_{sc}	Solar constant (1361 W/m²)
E_0	Eccentricity correction factor
τ_{ext}	Atmospheric extinction optical depth
m	Air mass (relative optical path length)
ϕ	Latitude (in radians)
δ	Solar declination angle (in radians)
θ_z	Solar zenith angle (in degrees)
h	Hour angle (in radians)
$\tau_{rayleigh}$	Rayleigh scattering optical depth
$\tau_{aerosol}$	Aerosol optical depth
τ_{ozone}	Ozone absorption optical depth
τ_{gases}	Gaseous absorption optical depth
$\tau_{water\ vapor}$	Water vapor absorption optical depth
k_{ozone}	Ozone absorption coefficient
k_{water}	Water vapor absorption coefficient
u_{ozone}	Total ozone column amount (atm·cm)
u_{water}	Precipitable water vapor (cm)
T_k	Temperature in Kelvin
T_{avg}	Average annual temperature at sea level (°C)
T_{base}	Base temperature at sea level (°C)
A	Amplitude of temperature variation (°C)
lapse rate	Temperature lapse rate (°C/m)
Height	Altitude above sea level (m)
P	Local atmospheric pressure (Pa)
P_0	Standard atmospheric pressure (101325 Pa)
λ	Wavelength (in micrometers)
β	Aerosol turbidity coefficient
α	Ångström exponent
g	Gravitational acceleration (9.80665 m/s²)
M	Molar mass of air (0.0289644 kg/mol)
R	Universal gas constant (8.31432 N·m/(mol·K))
DNI	Direct Normal Irradiance (W/m²)
TMY	Typical Meteorological Year
SAM	System Advisor Model
GUI	Graphical User Interface
Python	High-level programming language used in computational modeling