Forecasting the power output of photovoltaic cells: a comparison of statistical, machine learning and deep learning models

ABSTRACT

Power outputs from photovoltaic cells have previously been forecasted using weather data and data from large-scale photovoltaic plants. These models are primarily used for short-term forecasting ranging from a few minutes to a few days into the future. This project analyses the effects of weather features on power output and compares three different types of models: SARIMAX, XGBoost and LSTM-RNN, developed using UK weather data and power output data from a UK residential photovoltaic system. The models are used for longer-term forecasting of one-year of power outputs and compared using plots, RMSE, MAE and R^2. All weather features in the data were found to have significant correlations to the power outputs. The best performing model overall was the LSTM-RNN from the RMSE and MAE.

Code is available at: https://github.com/AliWilkinson/Solar_panel_power_forecasting

1 Introduction

Solar Panels or Photovoltaic (PV) cells produce a renewable energy source through the conversion of light energy from the sun to electrical energy for use locally or exportation to the National Grid. With the effects of climate change increasing (The Causes of Climate Change, 2023); diminishing fossil fuel resources (Shafiee and Topal, 2009); and rising energy costs (Guan *et al.*, 2023), global energy production will become more reliant on renewable energy sources such as PV installations (Hu *et al.*, 2016). Worldwide, countries are turning to renewable energy sources to combat these issues. The UK government, for example, has "committed to achieving fully decarbonised electricity by 2035, subject to security". (HM Government, Department for Energy Security and Net Zero, 2023, p.27). This report mentions maintaining and establishing forms of renewable energy including solar energy and aims for a five-fold increase in current installed rooftop capacity.

Forecasting models are often built using data from large scale PV plants (Photovoltaic power station, 2023), rather than small residential PV systems (Vagropoulos *et al.*, 2016). Grid companies often need to adjust distribution during high and low production periods (Poti *et al.*, 2023), and if residential PV systems outputs are not forecasted, potential surges from these systems cannot be anticipated. Currently 4.1 % of homes in the UK generate electricity through solar panels (*The MCS Data Dashboard - MCS*., 2023), and with the government's goal of increasing this number, developing forecasting models using data from small scale PV systems is paramount to predicting the potential increases in energy production.

Many models are built using data from other countries with differing climates to the UK. Weather conditions are seen to affect PV system power output (Shadid *et al.*, 2023), therefore, there is a need to develop models incorporating UK weather data into the forecast predictions. Combining data from UK residential PV systems and local weather data provides novel input data for developing forecasting models. Additionally, this project focusses on longer term predictions as opposed to one-day-ahead forecasting. The review by Mellit *et al.* (2020) states few studies focus on long-term forecasting and expresses the need for its investigation.

Building these models not only aids in meeting the goals mentioned above, but will also have economic benefits for PV system companies and prosumers (individuals with solar panels on private buildings who produce, use, and sell electricity to the National Grid) (Gautier, Jacqmin and Poudou, 2018). User requirements are discussed further in chapter 5.

The aim of this project is to build and evaluate commonly used PV power output forecasting techniques and suggest a model for multivariate, long-term forecasting of a UK residential PV system power output.

Outline of project objectives:

- Analyse how the residential PV systems power output changes with time.
- Analyse weather features effects on power output.
- Incorporate target lags and weather data to build three long-term, multivariate forecasting models: statistical, machine learning and deep learning.
- Use predictions for twelve months to compare results using plots and performance metrics.

2 Literature review

Climate effect on power output

Changing climate conditions are thought to impact the renewable energy sector in the future (Liu *et al.*, 2023). More irregular weather conditions and increasing episodes of extreme weather will likely affect the consistency of PV systems power outputs (Feron *et al.*, 2021). Energy demand in different locations may also change. For instance, Northern countries could see a decrease in energy usage for heating in winter and an increase in summer for cooling systems due to rising global temperatures (Liu *et al.*, 2023; Jerez *et al.*, 2015), changing grid demands. Additionally, air pollution influences PV system output. Particulate matter from fossil fuel combustion can become deposited on the PV cells blocking solar radiation reaching the cells, thereby reducing the PV cells output (Song, Liu and Yang, 2021). With increasing global temperatures and humidity, this information is vital to predicting how PV systems may act and how solar energy systems may need to change in the future. Many of these studies also suggest that the renewable energy sector will unlikely be threatened by climate change, although renewable energy companies will need to adapt to changing supply and demand due to shifting weather patterns as well as dealing with reduced efficiency of the PV cells because of air pollution. Developing accurate forecasting models for power output prediction from PV installations allows monitoring of PV outputs.

Models

Many statistical models, machine learning (ML) and deep learning (DL) have been used for PV output forecasting (Table 1). Common statistical models include ARMA, ARIMA, SARIMA, SARIMAX models (Kim, Akhtar and Yang, 2023), and regression models (AlShafeey and Csáki, 2021). ML models comprise of Decision Trees (DT), Random Forest (RF), XGBoost (Didavi, Agbokpanzo and Agbomahena, 2021) and Support Vector Machines (SVM) (Shi *et al.*, 2012). A review found the most frequently occurring DL models to be: Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM-RNN) (Rajagukguk, Ramadhan and Lee, 2020).

DL methods are often shown to perform well. Verma *et al.* (2016) found an ANN model had better performance over linear regression (LR), logarithmic regression (Log R), and polynomial regression (PR). Kim, Akhtar and Yang (2023) developed a selection of traditional statistical forecasting methods and a DL method: Holt-Winters, Multivariate Linear Regression, ARIMA, SARIMA, ARIMAX, SARIMAX and LSTM-RNN. The LSTM-RNN produced the smallest error for short-term forecasting. Rajagukguk, Ramadhan and Lee (2020) compared multiple DL models and found LSTM to perform best out of standalone models. A hybrid convolutional neural network-LSTM (CNN-LSTM) outperformed the other models including the LSTM. LSTM-RNNs likely perform well in this field as they have capacity for the short-term and long-term memory. These aspects allow the network to remember patterns in data, including trends and seasonality. Lee and Kim (2019) found a LSTM-RNN outperformed both an ANN and a DNN. However, Sangrody *et al.* (2018) showed an ANN to outperform RNN variants. This may be due to Sangrody *et al.* (2018) using the targets moving-average instead of lags, possibly affecting the need for a memory parameter in RNNs and may be a reason for the ANN outperforming the RNNs.

Additionally, ML models have been developed for PV power output forecasting. Shi *et al.* (2012) developed four SVMs for one-day-ahead forecasting models based on historical power data and weather forecasts. Didavi, Agbokpanzo and Agbomahena (2021) compared DT, RF and XGBoost models, where XGBoost produced the smallest error. In the study by Phan, Wu and Phan (2021), a Kernel Principal Component Analysis-XGBoost model was developed using historical power

generation data and historical weather data as inputs. They found this model outperformed traditional XGBoost and PCA-XGBoost models.

Statistical models are also shown to be effective forecasters. Vagropoulos *et al.* (2016) showed SARIMAX, modified-SARIMAX and ANN had better forecast performances over a persistence model and SARIMA model for short-term forecasting. For intra-day forecasting the SARIMA model performed best.

Weather conditions as input features

Weather conditions are shown to impact PV systems power output (Shadid *et al.*, 2023; Ramli *et al.*, 2016) and as a result, weather data are often included as exogenous features in forecasting models (AlShafeey and Csáki, 2021). Including weather data in such models is seen to produce better performing models than univariate forecasting models, shown by AlShafeey and Csáki (2021) who compared models with different inputs. An ANN and multiple regression (MR) model were built for each input type, which included: weather data; a time-series of the power output lag data; and a combination of the two. Using the combination of inputs improved the performance of both the ANN and MR model.

Weather data has also been incorporated to other model types. Didavi, Agbokpanzo and Agbomahena (2021) compared three tree-based ML models: DT, RF and XGBoost using only weather data as inputs. As previous power output was not included, these models cannot be compared to other studies where lags of power output are included. Hence, the development of an XGBoost incorporating lag data and weather data is needed before comparison to other models.

Geographic location and PV power output

PV systems in many geographic locations, with different climates and weather conditions, have been used to develop forecasting models. Geographic location has been seen to be a factor affecting forecasting performance (Mellit *et al.*, 2020). Multiple studies have used PV systems based in tropical and subtropical climates such as India (Verma *et al.*, 2016), South China (Shi *et al.*, 2012), Benin (Didavi, Agbokpanzo and Agbomahena, 2021), Taiwan (Phan, Wu and Phan, 2021) and Hawaii (Shireen *et al.*, 2018). However, the UK has a maritime, temperate climate resulting in different PV system behaviour. Su, Batzelis and Pal (2019) used solar irradiance data, weather data and data from a PV station located in Norwich, England to develop a mix of ten DL and ML models. One hybrid model was also developed due to none of the models performing the best in all four seasons, this model had greater performance over the single method models. Of the ten models, the Non-linear Autoregressive Exogenous Neural Network (NARXNN) had the least error. From this, another study went on to develop a NARXNN model built using data collected over a four-year-period from PV stations in Kent and Lincolnshire (Fjelkestam Frederiksen and Cai, 2022). These studies compared the performance of DL and ML models; but did not build any statistical models such as ARMA based models or regression models using UK data. They also used Horizon times of 7 days or less.

Horizon time

The forecast horizon refers to length of time for which the model makes predictions (Rajagukguk, Ramadhan and Lee, 2020). Horizons are split by length: very short-term (several minutes ahead); short-term (one hour to one week ahead); medium-term (one month to one year ahead); and longer-term (one to several years ahead) (Raza, Nadarajah and Ekanayake, 2016). The majority of previous studies focus on very short-term or short-term forecasting due to better model performance being associated with shorter horizon times (Lipperheide, Bosch and Kleissl, 2015).

Limitations of previous models

DL models appear to consistently have greater performance compared to other models (Kim, Akhtar and Yang, 2023; Su, Batzelis and Pal, 2019). However, development of ML models with the inclusion of both temperate climate weather data and power output lags as input features is lacking. There also lacks the development of statistical models, such as ARMA-based models with data from a PV system located in the UK. Therefore, further development and comparison of the performances of statistical, ML and DL models using such data is needed. Due to the current lack of studies that focus on longer term forecasting, a horizon time of one year is used in this project. Although DL models have good performances, advantages of the statistical models such as the ARMA-based models, regression, and certain machine learning models like DT, is their greater interpretability than DL models. Additionally, the performance of a DL model is often dependent on large datasets, which may not always be available for smaller PV systems, reiterating the need for a comparison using such data.

Paper	**Models developed**	**Inputs**	**Best model/s (if applicable)**	**Accuracy**
Shi *et al.* (2012)	SVM	Weather forecasts, POD		RMSE MRE
Verma *et al.* (2016)	LR, Log R, PR, ANN	WD POD	ANN	Error per 100kWh Mapping percentage
Vagropoulos *et al.* (2016)	SARIMA, SARIMAX, modified SARIMAX and ANN	WD POD	ANN and a combined SARIMA and ANN model	nRMSE
Sangrody *et al.* (2018)	ANN, SVR, RNN, KNN, GRNN and GPR	WD, Month number, Moving average of target	ANN	MAPE
Lee and Kim (2019)	ANN, ARIMA, SARIMA, DNNs, LSTMSs	WD, Seasonal Factors	LSTM	MAE RMSE
Su, Batzelis and Pal (2019)	BPNN, BPNN with genetic algorithm optimisation, ENN, GRNN, ANFIS, NARXNN, KNN, ELM, RF, SVR, hybrid model	Solar irradiance data, WD, POD	Hybrid model and NARXNN	nRMSE
Jung *et al.* (2020)	LSTM	WD		RMSE nRMSE MAPE R^2
AlShafeey and Csáki (2021)	MR, ANN	WD POD Separately and together	ANN	MAE MSE RMSE
Didavi, Agbokpanzo and Agbomahena (2021)	DT, RF, XGBoost	WD	XGBoost	MSE Regression value
Phan, Wu and Phan (2021)	XGBoost	WD		RMSE

Fjelkestam Frederiksen and Cai (2022)	NARXNN	WD POD		nRM
Essam *et al.* (2022)	ANN, RF, DT, XGBoost, LSTM	WD POD	ANN > RF > DT > XGB > LSTM	MAE RMSE R^2
Kim, Akhtar and Yang. (2023)	Holt-Winters, Multivariate Linear Regression, ARIMA, SARIMA, ARIMAX, SARIMAX, LSTM	WD POD	LSTM	MAE MSE RMSE R^2
Dhaked, Dadhich and Birla (2023)	LSTM, BPNN	WD POD	LSTM	MAE MAPE RMSPE R^2

Table 1: A summary of research papers that develop and compare forecasting models.
WD – Weather Data.
POD – Power Output Data.
See appendix **A** for all abbreviations.

Datasets

PV system data: from a residential installation located in Southwest UK; contained the power generated from the system for each month in kilowatt hours (kWh).

Weather data: obtained from the Met Office historic station data archive (*Historic station data - Met Office*, 2023). Yeovilton station was closest to the PV system (~20 miles).

There was no missing data for the period over which the solar panel power generation was measured.

Ethics

Permission to use the PV system data in this project is granted by the owner. Weather data is published under the Open Government Licence version 3.0. No personal identifying information or sensitive information is present in the data, so there are no ethical implications for using the data. Some methods can use lots of processing power, although this is limited in this project by the small dataset size.

3 Methods

Figure 1 shows a graphical overview of the technical processes used for data preprocessing and model building.

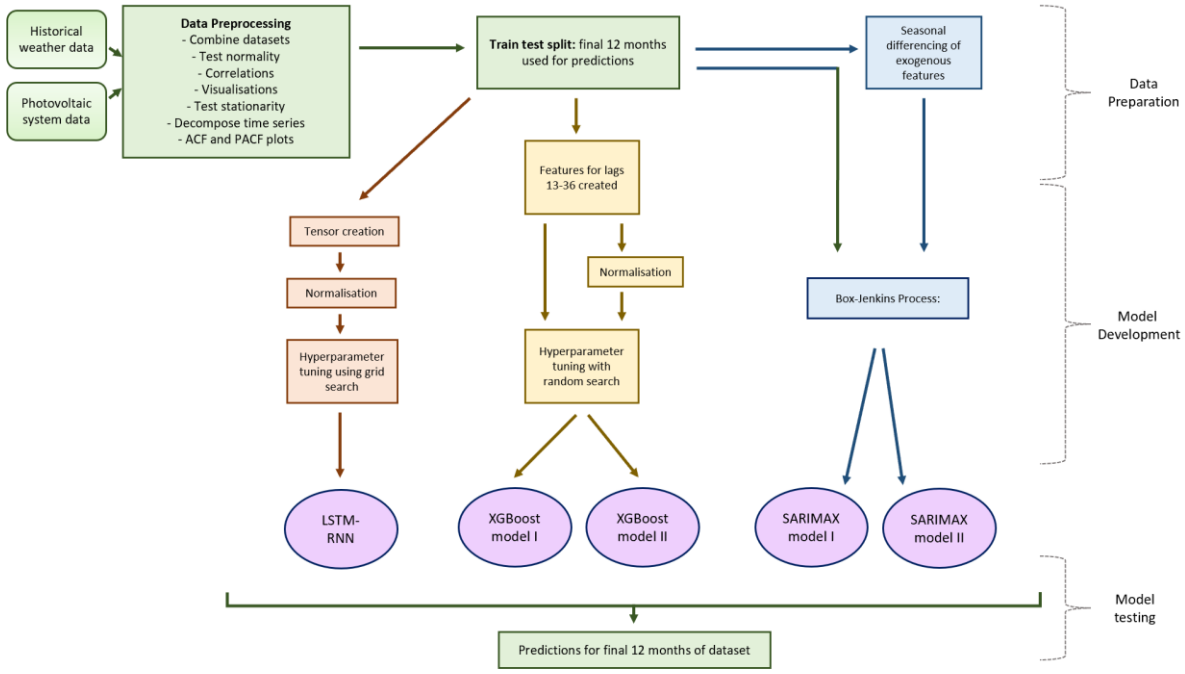

Figure 1. Graphical representation of technical processes.

Data preparation

The two datasets were combined. Python and libraries: Pandas (*pandas - Python Data Analysis Library*., 2023), Matplotlib (Hunter *et al.*, 2023) and Seaborn (Waskom, 2021) were used for data manipulation and visualisation.

In the following sections, variables are referred to as they are named in the code:

- *Monthly Power Generation*.
- *Sun Hours*: total monthly sunshine duration.
- *Tmax degC*: mean daily maximum temperature (°C).
- *Tmin decC:* mean daily minimum temperature (°C).
- *Mean Monthly Temperature:* created from *tmax degC* and *tmin degC.*
- *Rain mm*: total rainfall for the month (mm).
- *No. Frost Days:* monthly days of air frost.

The distribution of each variable was analysed. The target variable (*Monthly Power Generation*) was tested for a normal distribution using the Shapiro-Wilk test.

The null hypothesis (H_0): the data is normally distributed.

When the p-value > 0.05, the H_0 is failed to be rejected (Mishra *et al.*, 2019). The target variable was shown not to be normally distributed, therefore Spearman's correlation was used over Pearson's correlation to observe the correlations between the variables as Pearsons requires normally distribute data.

The variables *tmax degC* and *tmin degC* were strongly correlated as they both describe temperature; hence the variable *Mean Monthly Temperature* was created (see Equation 1 for calculation). Having highly correlated exogenous variables leads to multicollinearity, which negatively affects model performance, by causing overfitting. The new variable was also created to reduce the number of variables compared to the amount of data. As this dataset only contains 133 total observations, having many variables could lead to overfitting models. Feature engineering aids in combating the curse of dimensionality (Altman and Krzywinski, 2018).

$$[1] \qquad (tmax\ degC + tmin\ degC)/2$$

Plots of the target variable against each exogenous variable were plotted to view relationships, and the significance of the Spearman's correlation for each variable to the target variable was tested by comparing the p-value to significance level of 0.05.

The statsmodels library seasonal_decompose function (Perktold *et al.*, 2023) allows the breakdown of a time series to its trend, seasonal component and residuals. This can be used to identify stationarity and seasonality. In stationary time series, the time of the observation does not affect the value (Hyndman and Athanasopoulos, 2018). SARIMAX models require data to be stationary to make forecasts. They contain parameters (orders) that define how many times to difference or seasonally difference the data. These can be investigated by statistical tests for stationarity and by plotting the autocorrelation (ACF) and partial autocorrelation (PACF).

Variables were decomposed, and the Augmented Dickey-Fuller (ADF) test was used to examine if the features were stationary. The H_0 and the alternative hypothesis (H_A) for the ADF test were defined as:

H_0: The time series is non-stationary due to trend.
H_A: The time series is stationary.

If p-value < 0.05, the H_0 is rejected. The ADF test was performed for the target variable with differencing and seasonal differencing to observe the effects on the stationarity of the time series.

ACF and PCAF plots were produced for the target variable, seasonally differenced target variable and the exogenous features. ACF plots show the correlation between the lag values of a time series and PCAF show the correlation between the lag values of a time series with the effects of previous lag removed. They can allow for patterns in the correlations of lags to be viewed, such as seasonal patterns. They can also be used to reveal the orders of AR and MA models.

The data was split into training and testing sets to allow the final twelve months of the data to be used to test and compare model performance.

Performance metrics

The RMSE (Hodson, 2022), equation 2, was used as the primary metric to measure model performance. In RMSE, larger residuals are punished over smaller ones and residuals' the sign has no effect. RMSE is the most frequently used model evaluation metric in forecasting photovoltaic cell output (Rajagukguk, Ramadhan and Lee, 2020).

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

The mean absolute error (MAE) was also measured, equation 3, it gives an absolute difference between actual and predicted values.

$$[3] \qquad MAE = \frac{\sum_{i=1}^{n} |\hat{y}_i - y_i|}{n}$$

The R^2 value was also measured, equation 4, it evaluates the goodness-of-fit of the model.

$$[4] \qquad R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y}_i)^2}$$

A combination of metrics gives a thorough evaluation of performance. These metrics were chosen as they often appear in studies that evaluate PV output forecasting models, Table 1.

For long-term forecasting, a horizon time of one year was used.

Forecasting Models

SARIMAX

Seasonal Autoregressive Integrated Moving Average with exogenous factors (SARIMAX) is an extension of an ARIMA model allowing for seasonal data and the addition of exogenous features into the model.

The SARIMA model: ARIMA (*p, d, q*) (*P, D, Q*)s. (*p, d, q*) are the orders of the non-seasonal section of the model and the (*P, D, Q*)s are the orders of the seasonal section.

p is the order of the autoregressive model ($AR_{(p)}$) and refers to the number of lags used in the model. Time series values often have relationships with previous values in the time series, also known as autocorrelation. AR models allow lag values of the target variable to be incorporated into the model. An $AR_{(p)}$ model is expressed in equation 5.

$$[5] \qquad y_t = c + a_1 y_{t-1} + a_2 y_{t-2} + \cdots + a_p y_{t-p} + \epsilon_t$$

Where y_t is the target variable, ϵ_t is white noise, c is a constant and $a_1 \ldots a_p$ are coefficients. The coefficients determine the pattern of the time series.

d is the order of differencing, referring to the number of times the data is differenced to achieve stationarity, equation 6.

$$[6] \qquad y'_t = y_t - y_{t-d}$$

q is the order of the moving average model ($MA_{(q)}$) and refers to the number of lags used in the model. The MA section of the model regresses the time series values against the errors of predictions of past values. An $MA_{(q)}$ model is expressed in equation 7.

$$[7] \qquad y_t = c + m_1 \epsilon_{t-1} + m_2 \epsilon_{t-2} + \cdots + m_q \epsilon_{t-q} + \epsilon_t$$

Where c is a constant, ϵ_t is white noise and $m_1 \ldots m_q$ are the coefficients which determine the pattern of the time series.

P: order for the seasonal autoregressive part of the model.

D: number of times differencing is required to remove the seasonal component of the data. The value from one cycle ago is subtracted from the current value, equation 8.

$$[8] \qquad y'_t = y_t - y_{t-s}$$

For monthly data with a seasonal period of twelve months, s =12. So, for D = 1, $y_t' = y_t - y_{t-12}$.

Q: order for the seasonal moving average part of the model.

s: number of data points the seasonal component repeats over.

Exogenous features are also incorporated into the model. The SARIMAX model is mathematically defined as equation 9 (Vagropoulos *et al.*, 2016).

$$[9] \qquad \phi_p(B)\Phi_P(B^S)\nabla^d\nabla_S^D y_t = \beta_k x_{k,t}{}' + \theta_q(B)\Theta_Q(B^S)\epsilon_t$$

B refers to lag values i.e. when B is applied to y_t, it gives y_{t-1} (Hyndman and Athanasopoulos, 2018).

The components are defined as:

- $x_{k,t}$ each explanatory variable (k^{th} variable) at time t with the coefficient β_k.
- y_t the forecast variable at time t.
- $\phi_p(B)$ refers to the AR model with an order of *p.*
- $\theta_q(B)$ refers to the MA model with an order of *q*.
- $\Phi_P(B^S)$ refers to the seasonal AR model with an order of *P*.
- $\Theta_Q(B^S)$ refers to the seasonal MA model with an order of *Q*.
- ∇^d refers to the differencing.
- $\nabla^D{}_S$ refers to the seasonal differencing.
- ϵ_t white noise.

(Vagropoulos *et al.*, 2016)

The Box-Jenkins process (Box, Jenkins and Reinsel, 2008) was followed to find the parameters and build the SARIMAX model. The steps of the Box-Jenkins process are outlined in Figure 2.

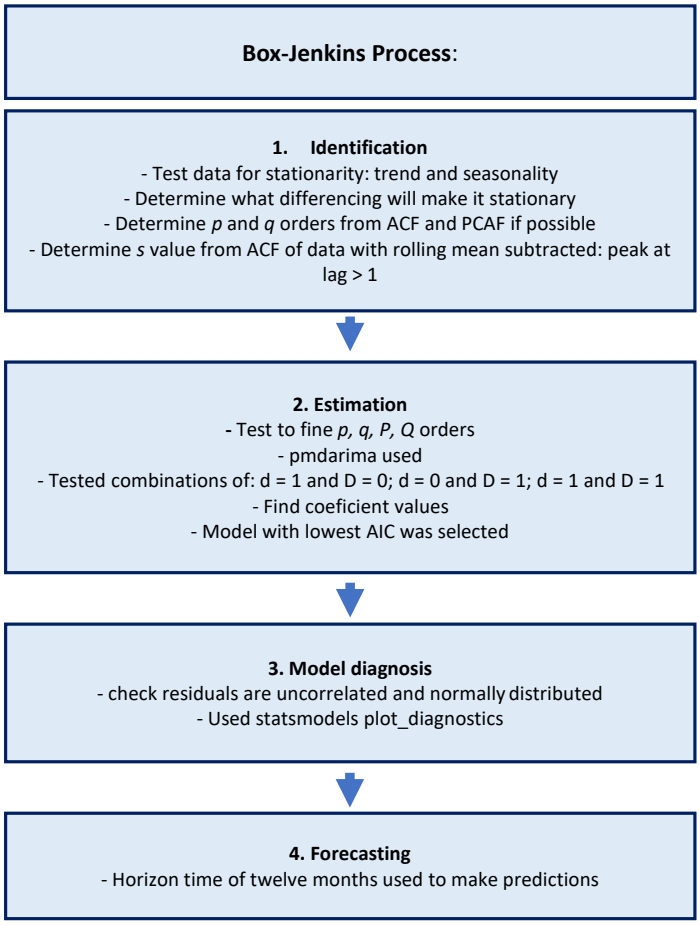

Figure 2. Flow diagram illustrating the Box-Jenkins process.
pmdarima (*pmdarima: ARIMA estimators for Python — pmdarima 2.0.4 documentation*, 2023).
ACI - Akaike's Information Criterion used as the model being developed is a forecasting model.
statsmodels plot_diagnostics method (Perktold *et al.*, 2023).

A second SARIMAX model was developed to compare the effect of seasonally differencing exogenous features on the model performance and fit. Seasonal differencing was performed prior to model building and the same process as above was followed.

XGBoost (XBG)

XGBoost (extreme gradient boosting) is an ensemble ML method. DTs were used as the base learners, known as classification and regression trees in XGB. The general XGB formula is expressed in equation 10.

$$[10] \qquad \hat{Y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F$$

For K additive functions, F is a function of all trees and f represents individual trees (Chen and Guestrin, 2016). Each tree is built with the goal of minimising objective function in equation 11.

$$[11] \qquad Obj(f_1, \dots, f_k) = \left[\sum_i l\left(y_i, \hat{Y}_i\right)\right] + \sum_k \Omega(f_k),$$

$$\Omega(f) = \gamma T + 1/2\lambda \sum_{j=1}^{T} \omega^2$$

- $[\sum_i l(y_i, \hat{Y}_i)]$ is the loss function. For the regression trees, $½(y_i - \hat{Y}_i)^2$ was used; $\hat{Y}_i$ is the prediction and y_i is the target value.
- ω is the output values of the terminal node.
- λ is the regularisation term.
- γ is a parameter which encourages tree pruning.
- T is the number of terminal nodes in tree f.
- j is the individual terminal nodes.

XGB makes an initial estimate. The residuals between the true values and the prediction are measured. The first tree is then built using the input features to group residuals. Trees are built to minimise the loss function, see above. Output values for the tree are calculated, equation 12.

$$[12] \qquad \Omega = sum\ of\ residuals\ /\ (No. residuals + \lambda)$$

The regularisation term, λ, reduces the model sensitivity to individual observations, this helps prevent overfitting to the training data. The new predictions for each value are compared to the true values and the residuals measured, see equation 13.

$$[13] \qquad Predictions\left(\hat{Y}_i\right) = previous\ prediction + (learning\ rate * \omega)$$

The process then repeats with the residuals of the new tree until very small residuals are produced or the pre-set max number of trees is reached. Trees are added to the ensemble with the goal of minimising the objective function.

The XGBoost library (*XGBoost Documentation — xgboost 2.0.2 documentation*., 2022) was used to create an XGB model. A random search (sklearn.model_selection.RandomizedSearchCV, 2023) was used to tune the model's hyperparameters. Table 2 shows the hyperparameters that were tuned and what they represent.

Hyperparameter	Hyperparameter meaning
n_estimators	Maximum number of trees included in the model.
max_depth	Maximum depth for trees. Larger value = a more complex model.
learning_rate	The value that the weights of the new features are shrunk by. Larger value = more conservative model.
gamma	Minimum loss reduction for the tree to make another split. Larger value = more conservative model.
lambda	Regularisation term. Larger value = more conservative model.
colsample_bytree	The percentage of input features used for a given tree. 0-1. Large values can cause overfitting.

Table 2. Hyperparameters tuned in XGB model and their meanings. (*XGBoost Parameters — xgboost 2.0.2 documentation*., 2022)

To incorporate lags of the target variable and exogenous variables into the model, all lag values up to lag thirty-six were calculated for each feature. As the data were monthly values and the goal was to

forecast one year of power outputs after the end of the training data, the first twelve lags needed to be removed from the training and testing set. This is because XGB uses the values of the exogenous features for each observation to build trees and make predictions. The testing set is treated as a period in the future where no values are known for any of the variables. Therefore, to use any present values of the exogenous features or any lags that correspond to other observations within the testing set would give a heightened performance of the model. Whereas, in the SARIMAX models, lag values are incorporated by the AR and MA sections of the models and predictions are made using the forecast function, that does not require the input of an X_test dataset.

LSTM RNN model

The Long Short-Term Memory Recurrent Neural Network (LSTM-RNN) model is an ANN. ANNs are based on the structure of neurons in the brain, they are made up of layers which contain 'neurons': these contain nodes connected by weights. The basic structure of ANNs is one input layer, hidden layer(s) and one output layer, Figure 3.

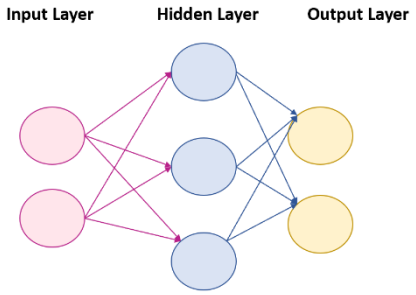

Figure 3. Basic structure of an ANN

Layers are connected by weights. During forward propagation, input values move through the network, input and weight are multiplied, and a bias value is added. Backpropagation uses gradient descent to calculate optimal values for the weights and biases starting with the last parameter and working backwards.

RNNs are a type of ANN that allow backward pointing connections within the network. This permits information about the previous input, known as the hidden state, to be carried forward and combined with the next input to make predictions. This allows the network to have memory, making it useful for forecasting problems where predictions can be made from previous values. However, RNNs have very short-term memory and cannot include longer-term interactions.

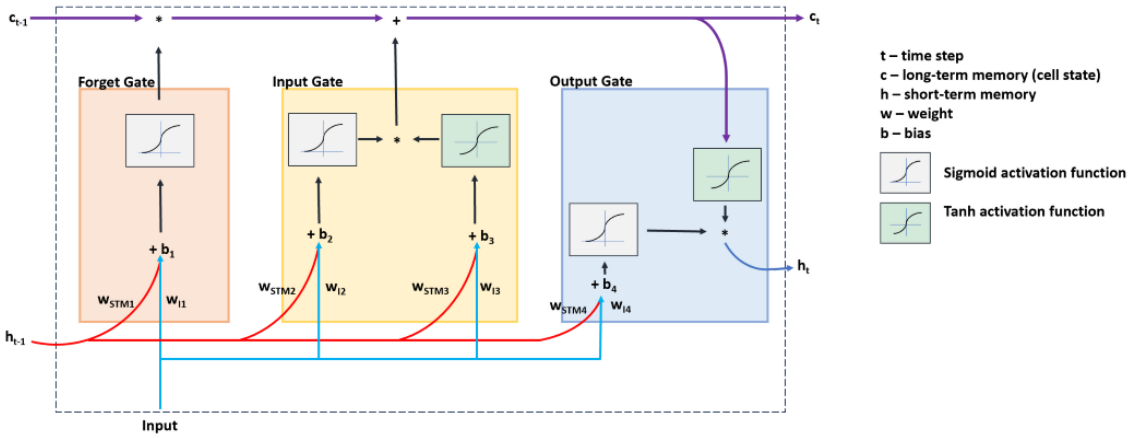

Figure 4. A LSTM cell showing which components interact through each gate. STM – short-term memory. I – Input.

LSTM cells are units that can be used in RNNs. They allow RNNs networks to capture longer-term relationships in sequential data, such as time series data. LSTM units also help prevent vanishing and exploding gradients; problems that can occur with RNNs (Sherstinsky, 2020).

Inside the LSTM cell there is a sequence of steps, Figure 4**.** In the forget gate, the input (I), short-term memory (STM), the input and STM weights (w_{I1} and w_{STM1}) and bias (b) are combined, equation 14.

$$[14] \qquad (I * w_{I1}) + (STM * w_{STM1}) + b_1$$

The result is then inputted into the sigmoid activation function (SAF) to determine what percentage of the long-term memory (LTM) is to be passed to the next step. The output from the SAF is multiplied to the LTM.

The input gate creates a potential LTM by the equation 15.

$$[15] \qquad (I * w_{I3}) + (STM * w_{STM3}) + \ b_3$$

This is inputted to a tanh activation function (TAF); the output is a potential LTM.

Simultaneously, equation 16 is performed and the result passed to the SAF. The output determines how much potential LTM is retained and added to the updated LTM from the forget gate.

$$[16] \qquad (I * w_{I2}) + (STM * w_{STM2}) + b_2$$

Finally, in the output gate, the TAF takes the updated LTM from the input gate and produces potential STM.

The result from equation 17 is passed to the SAF.

$$[17] \qquad (I * w_{I4}) + (STM * w_{STM4}) + b_4$$

The output from the SAF is multiplied to the potential STM to produce the output of the LSTM cell (h_t).

Pytorch (*PyTorch documentation — PyTorch 2.1 documentation*., 2023) was used to build the neural network. This library was chosen for its compatibility with Python and fast performance (Pykes, 2023).

Normalisation was performed as neural networks are sensitive to feature scales. No differencing was performed as networks can be used with complex, non-stationary data (Inman, Pedro and Coimbra, 2013). A simple LSTM-RNN model was built with a LSTM layer and one linear layer. The linear layer takes the hidden size as its input and produces the number of outputs of the prespecified output size. In regression problems, the final layer must be a fully connected linear layer. Hyperparameter tuning was performed using a grid search. Table 3 gives a description of the hyperparameters used in the model.

Hyperparameter	**Hyperparameter meaning**
Batch_size	The number of samples put to network.
max_epochs	Maximum number of iterations through the network
Hidden_size	The STM and the size of the output from the LSTM unit.
Num_layers	Number of LSTM layers
Learning_rate	Used in backpropagation to update weight parameters
Input_size	Number of features.
Output_size	Number of outputs from network.

Table 3. Hyperparameters tuned in the LSTM-RNN model and their meanings.

Input and output sizes were constant at five and twelve respectively. The model was trained with the best hyperparameter values from the tuning.

Adam optimiser was used for gradient descent during backpropagation. Adam was chosen as it is shown to perform well in neural networks, often out-performs other optimisers, such as stochastic gradient descent (Kingma and Ba, 2015) and is often the recommended optimiser (Ruder, 2017).

MSE loss was used as the loss function equation 18. This was used in the hyperparameter tuning and training process to measure model fit. In MSE loss, positive and negative differences between actual and predicted values are treated the same.

$$[18] \qquad MSE = 1/n \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

A lookback period of twenty-one data points was used.

4 Results

Exploratory Data Analysis

The data contained 133 rows, representing each month. The combined solar panel data and historical weather data contained no duplicated rows and no missing values for the variables of interest. Figure 5 shows the distribution of each variable (a, c, e, g, i and k) and a box plot for each variable (b, d, f, h, j and l).

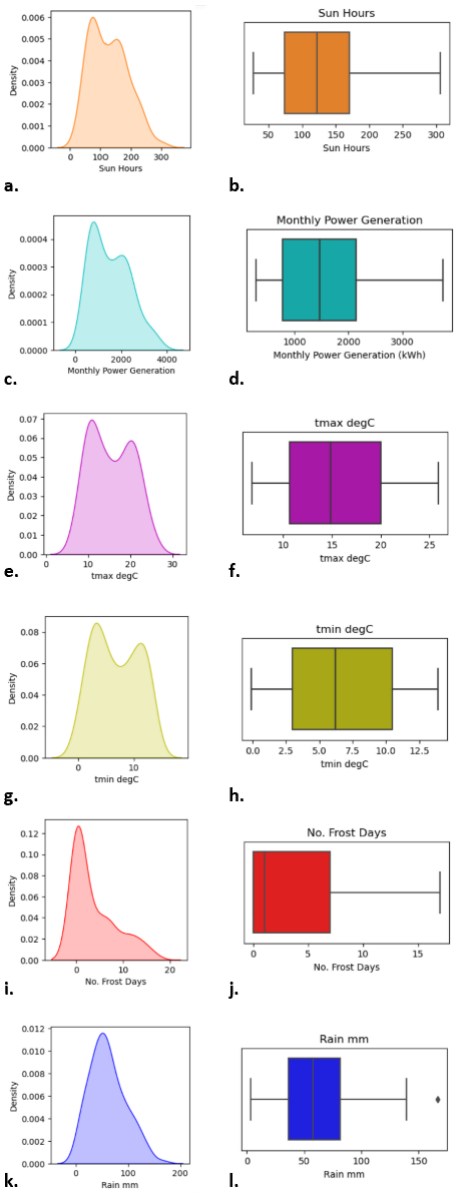

Figure 5. Distributions for each variable shown in the density plots and box plots.

Figure 5i shows there is an outlier present in the rain variable. This was retained as it was a true value and not a mistake in data collection.

The result from the Shapiro-Wilk test was 0.9470 with a p-value of $5.478e^{-5}$, therefore, the *Monthly Power Generation* is shown not to be normally distributed.

The heatmap, Figure 6, shows the Spearman's correlations (ρ) between the variables. The features *tmax degC* and *tmin degC* have a strong positive correlation (0.95).

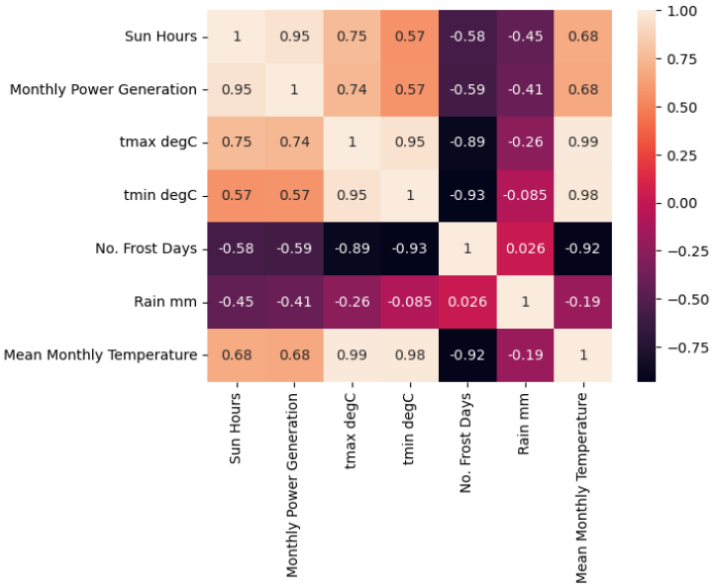

Figure 6. Heatmap showing Spearman's correlation between each variable. Included: original weather variables and created variable: *Mean Monthly Temperature*.

Sun hours has the strongest correlation with *Monthly Power Generation* (0.95). *Mean Monthly Temperature* has a moderate positive correlation to *Monthly Power Generation*. *No. Frost Days* and *Rain mm* have moderate negative correlations to the *Monthly Power Generation*. These correlations would suggest relationships between the variables and the target, these are displayed in Figure 7.

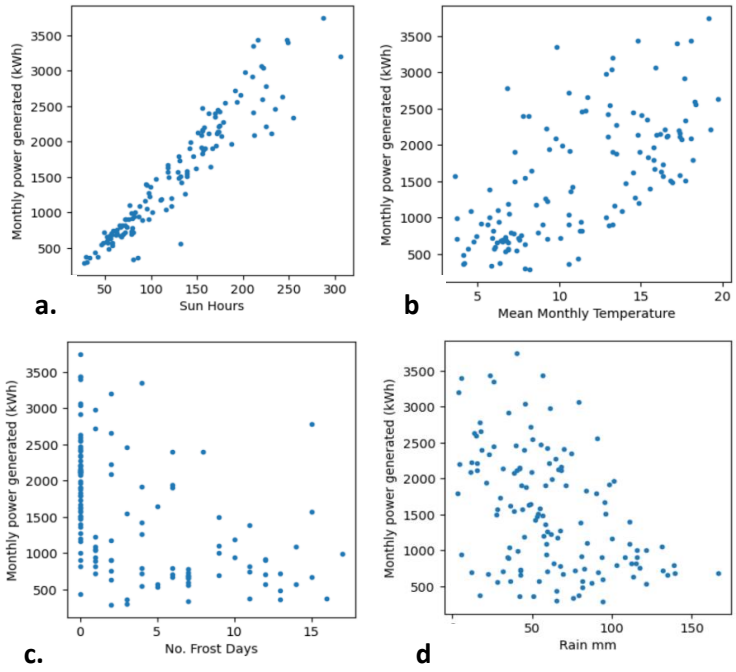

Figure 7. Plots of Monthly Power Generation (kWh) against each variable.

From Figure 7a the strong correlation between *Sun Hours* and the *Monthly Power Generation* is clear. 7b displays the relationship between *Mean Monthly Temperature* and *Monthly Power Generation*. From 7c and 7d it is harder to see a definitive relationship between the variables *No. Frost Days* or *Rain mm* and the *Monthly Power Generation*. To confirm the statistical significance of the correlations, p values were calculated are shown in Table 4**.** As all the correlations have a p-value < 0.0001, they are statistically significant and so are maintained and used in forecasting models.

	Spearman's Correlation	p-value
Sun Hours	0.9506	0.000000
Mean Monthly Temperature	0.6751	0.000000
No. Frost Days	-0.5853	0.000000
Rain mm	-0.4069	0.000001

Table 4. Values of the Spearman's Correlation for each variable with *Monthly Power Generation* and the corresponding p-value.

Figure 8 shows the *Monthly Power Generation* for the whole dataset and how it changes over time. The rolling mean (blue) and rolling standard deviation (orange) show a downward trend in the amount of power generated per month. There also looks to be a seasonal pattern in the data

repeating each year. This is concurrent with knowledge on photovoltaic cell power generation being affected by seasons and weather conditions.

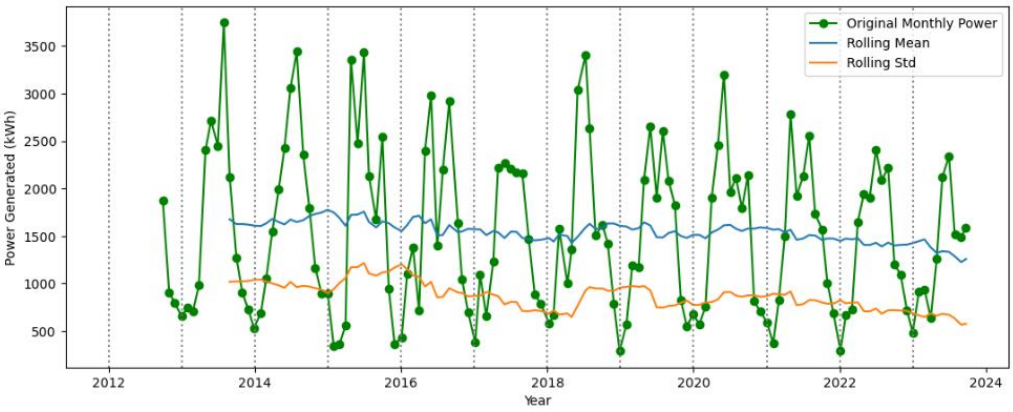

Figure 8. Line plot of the original Time Series: *Monthly Power Generation* over time (green). With the rolling mean (blue) and the rolling standard deviation (orange).

Stationarity and Seasonality

To establish if the features and the target are stationary, each feature was decomposed, this is shown in Figure 9**.** Figure 9a, c, e, g and i show the plotted time series and b, d, f, h, and j show the corresponding decomposed variables. The target variable, *Monthly Power Generation* (9a and 9b), is shown to have a downward trend and strong seasonal component. The decomposed *Mean Monthly Temperature* (9f) may have a slight upward trend. The other exogenous features do not show any clear trends. The features *Sun Hours* (9d), *Mean Monthly Temperature* (9f) and *No. Frost Days* (9j) display clear seasonal components.

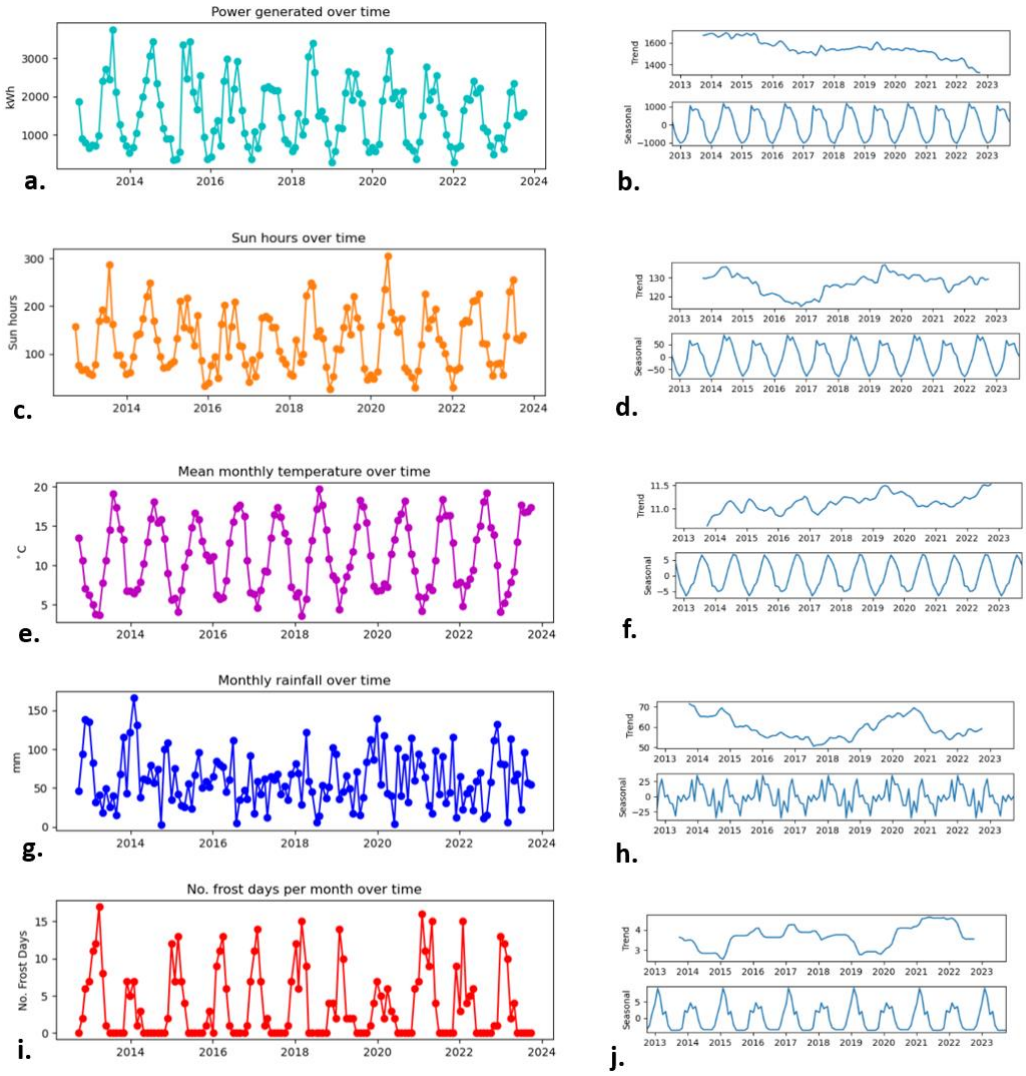

Figure 9. Right: time series for each variable and the target variable. Left: decomposition results (trend and seasonality) corresponding to the variable on the right.

Table 5 shows that the *Monthly Power Generation* variable is not stationary without the data being differenced. It also shows that one differencing and one seasonal differencing made the data stationary.

	ADF p-value
No differencing	0.884
Differenced once	0.000 ($3.219e^{-12}$)
Seasonally difference once	0.006

Table 5. p-values for the ADF of the *Monthly Power Generation* variable with no differencing, differenced once and seasonally differenced once.

The exogenous features were tested for stationarity, Table 6**.** All exogenous features are stationary according to the ADF test, meaning there is no significant trend present in any of the exogenous features.

	ADF p-value
Sun Hours	0.0000
No. Frost Days	0.0000
Rain mm	0.0000
Mean Monthly Temperature	0.0074

Table 6. The p-value from the ADF test for each exogenous feature.

ACF plots were also created for each exogenous feature to view autocorrelation. Figure 10a, c and d show clear correlations between the lags of the features: *Sun Hours, No. Frost Days* and *Mean Monthly Temperature*. Strong negative correlations at lag six can be seen for all these features and strong positive correlations can be seen at lag twelve. This shows the presence of a yearly season structure in the data. From Figure 10b and Figure 9h there appears to be no seasonal component or trend in the *Rain mm* feature.

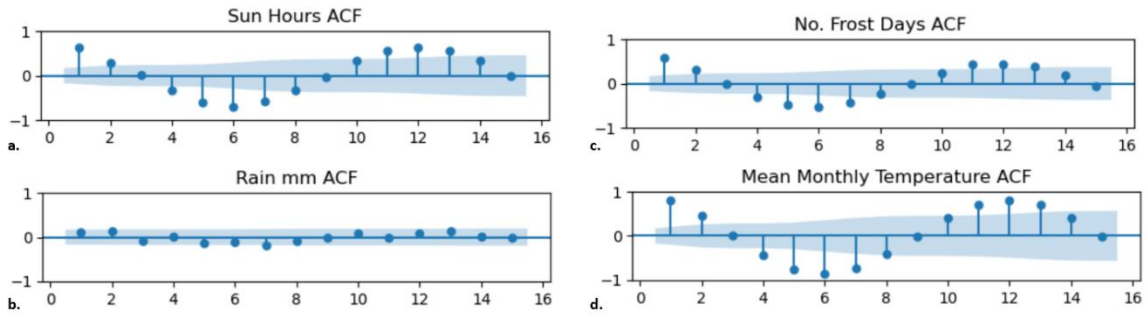

Figure 10. ACF plots for each exogenous feature.

The ACF plot for the detrended *Monthly Power Generation*, Figure 11**,** shows how many data points the seasonal component repeats over. A peak outside of the blue region, greater than zero, at lag twelve means the values have statistically significant correlations to the observation twelve data points prior, indicating a yearly seasonal component. This coincides with the line graph in Figure 8 and knowledge that photovoltaic cell output is dependent on weather conditions.

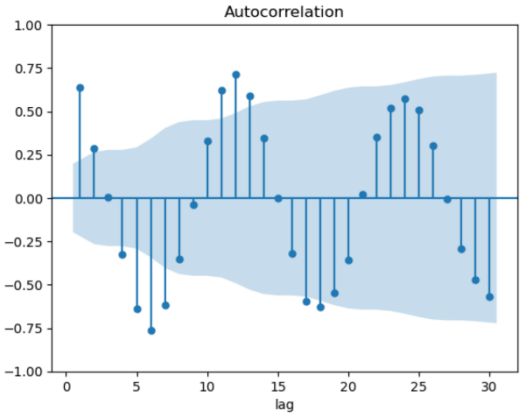

Figure 11. ACF plot for detrended *Monthly Power Generation.*

Train Test Split

Figure 12 shows how the data was split for models. The blue section represents the data used to train the models. The orange section represents the final twelve months which were compared to the model predictions to measure model performance.

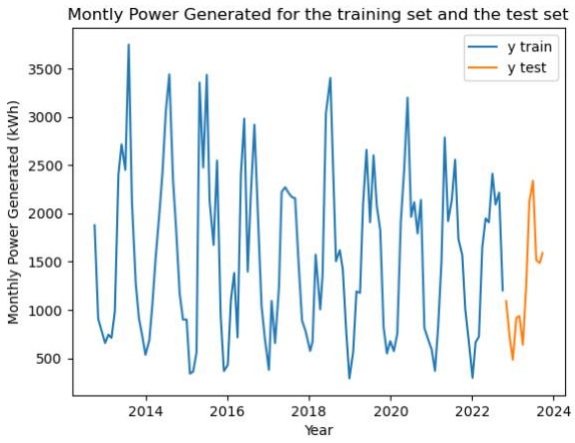

Figure 12. The training (blue) and test split of the data (orange).

Model Results and Performance

ACF and PACF plots can be used to identify the order of the AR and MA components of SARIMA models. The ACF and PCAF plots for the seasonally differenced target are plotted, Figure 13. However, the orders were unable to be identified from the plots and other techniques were used to identify these.

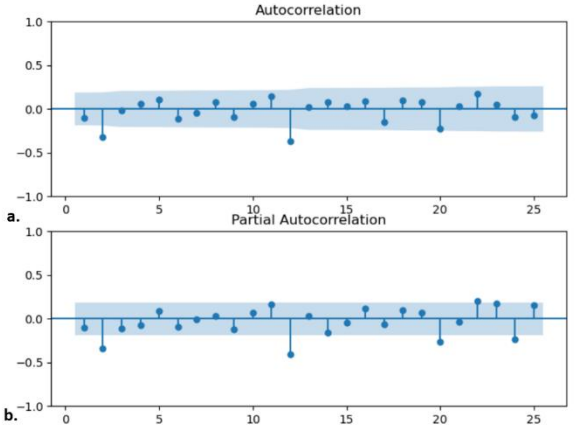

Figure 13. ACF and PACF plots for the seasonally differenced *Monthly Power Generation*. Twelve data points is the seasonal period (see Figure 11).

For the first SARIMAX model, an ARIMA(2, 1, 2)(2, 1, 1)$_{12}$ gave the smallest AIC value out the models tested. Table 7 shows the best model produced with different combinations of d = 0, d = 1, D = 0 and D = 1.

Model	Order	Seasonal order	AIC
ARIMA(2,1,2)(2,1,1)[12]	(2, 1, 2)	(2, 1, 1, 12)	1638.254747
ARIMA(2,0,0)(2,1,1)[12] intercept	(2, 0, 0)	(2, 1, 1, 12)	1645.817225
ARIMA(3,1,0)(1,0,1)[12] intercept	(3, 1, 0)	(1, 0, 1, 12)	1873.186140

Table 7. The best models produced from each run of the SARIMAX model building process. Models are sorted in order of AIC (lowest-highest).

The summary of the SARIMAX model after fitting to the training data gave the statistics and p-values of the Ljung-Box test and Jarque-Bera test. The Ljung-Box test is used to test autocorrelations of residuals. The H_0 is the residuals have no correlation to each other. The Jarque-Bera tests for normality. The H_0 is the residuals are normally distributed. A p-value < 0.05 for either test means the H_0 is rejected. The p-values of the Ljung-Box test and Jarque-Bera test were 0.88 and 0.00 respectively.

The plots in Figure 14 show the model diagnostics and analyse the of the goodness of fit of the model. Figure 14a shows that there is no structure to the residuals. 14b shows a histogram and distribution of the residuals compared to a Gaussian normal distribution. There appears to be a slight negative skew in the distribution of the residuals. If the residuals are normally distributed, most points should lie along the red line, Figure 14c. The correlogram, 14d, shows the autocorrelation of the residuals, lags after lag zero should not be significant (should lie inside the blue area). This shows there is no correlation between residuals.

From the results of the Ljung-Box test, Jarque-Bera test and the model diagnostics plots, Figure 14, the model residuals for the first SARIMAX model are shown not to have correlation to each other but are not normally distributed.

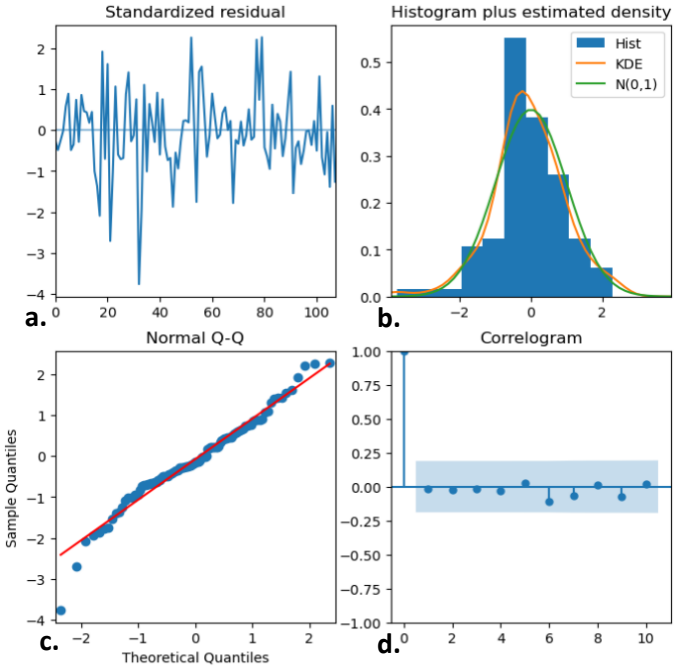

Figure 14. Model diagnostics plots of the SARIMAX model.

The RMSE value for this SARIMAX model was 429.7 kWh for the twelve-month horizon time. As the features: *Sun Hours, No. Frost Days* and *Mean Monthly Temperature* show seasonality. Seasonal differencing was performed on these variables and the SARIMAX model development process was repeated to include the seasonally differenced exogenous features to view if this changed the model's performance or fit. An ARIMA(2, 1, 2)$(2, 1, 1)_{12}$ gave lowest AIC value. The Ljung-Box test returned a p-value of 0.87 and the Jarque-Bera test returned a p-value of 0.25. Figure 15 shows the model diagnostics plots for the second SARIMAX models. This shows the effect of seasonal differencing of exogenous features on the goodness of fit of the model. Figure 15a shows no structure to the residuals. 15b and 15c show a more normal distribution than the previous SARIMAX model. Figure 15d shows no autocorrelation in the residuals.

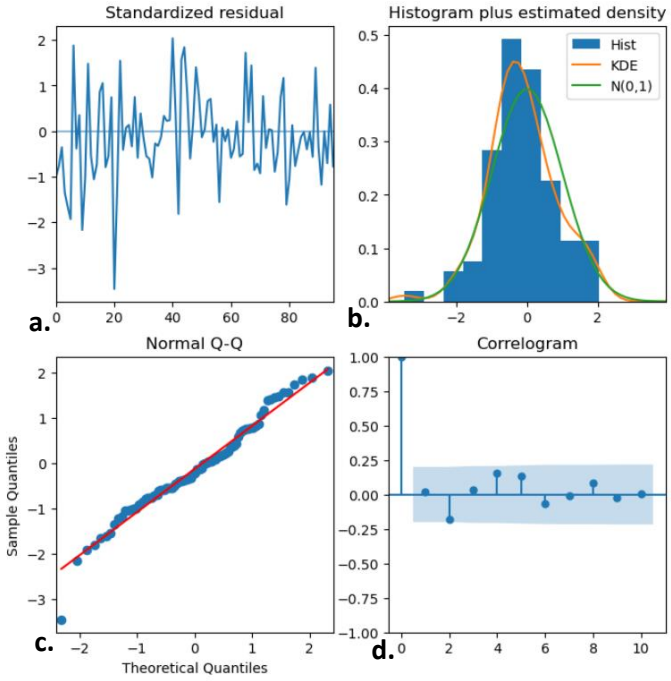

Figure 15. Model diagnostics plot of the SARIMAX model with seasonal differencing performed to exogenous features.

The RMSE value for the second SARIMAX model was 363.5 kWh for the test set. From the results of the models, results of the Ljung-Box and Jarque-Bera tests, and the model diagnostics plots, seasonal differencing on the exogenous features that displayed seasonality has improved model performance and model fit.

The hyperparameter values from the random search for the first XGB model are shown in Table 8.

Hyperparameter	Value
Number of trees	200
Max depth of tree	6
Learning rate	0.35
Lambda	3.3
Gamma	1.7
Percent of features used in trees	0.3

Table 8. The results of the random search for tuning the hyperparameters of the XGB model.

Normalisation was performed on the data and a second XGB model was developed. The hyperparameters for this model are shown in Table 9.

Hyperparameter	Value
Number of trees	500
Max depth of tree	5
Learning rate	0.35
Lambda	2.1
Gamma	0.0
Percent of features used in trees	0.3

Table 9. The results of the random search for tuning the hyperparameters of the second XGB model.

The results of the grid search approach used to tune the hyperparameters for the LSTM-RNN are shown in Table 10.

Hyperparameters	Value
Batch size	20
Max of epochs	100
Hidden size	64
Number of layers	1
Learning rate	0.001

Table 10. The results of the grid search for tuning the hyperparameters of the LSTM-RNN model.

Tabel 11 shows the RMSE, MAE and R^2 value for each model. The LSTM-RNN produces the best result for the RMSE and the MAE. The SARIMAX II model has the best R^2 score, followed by the LSTM-RNN with a 0.01 difference.

Model	RMSE (kWh)	MAE (kWh)	R^2
SARIMAX (without seasonal differencing of exogenous features)	429.7	322.2	0.72
SARIMAX (with seasonal differencing of exogenous features)	363.5	310.6	**0.78**
XGBoost (without normalisation)	502.7	381.6	0.75
XGBoost (with normalisation)	323.2	235.2	0.67
LSTM-RNN	**267.6**	**219.9**	0.77

Table 11. RMSE, MAE and R^2 value for each model.

Figure 16 shows the True Monthly Power Outputs and Predicted Monthly Power Outputs for each model. Both SARIMAX models and the XGB model without normalisation have a similar pattern to their predictions that mimics the shape of the true values but overestimates the power outputs for the 04/23 – 06/23 and the 08/23. Normalisation appears to stop the XGB model overestimating the power output. The XGB II model gives close estimates of power output up to the 06/23, then starts to underestimate. LSTM-RNN model produces a curve to estimate the power output, following the pattern of the true values less closely than the other models, but gives more accurate predictions. Figure 17 shows linear regression for Predicted Monthly Power Outputs compared against the True Outputs. The R^2 values, Table 11, correspond to these. The more accurate the model predictions, the closer the points lie to the line. Models appear to make better predictions when the true power output is low, these low power values correspond to winter the months.

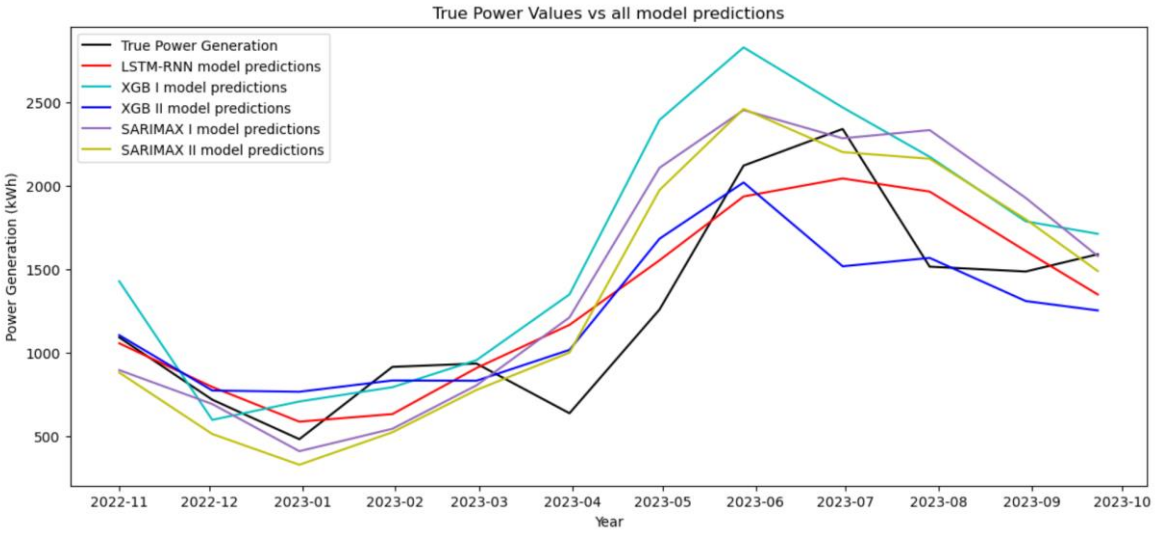

Figure 16. Plot of the final twelve months of true Monthly Power Outputs and the predictions from each model.

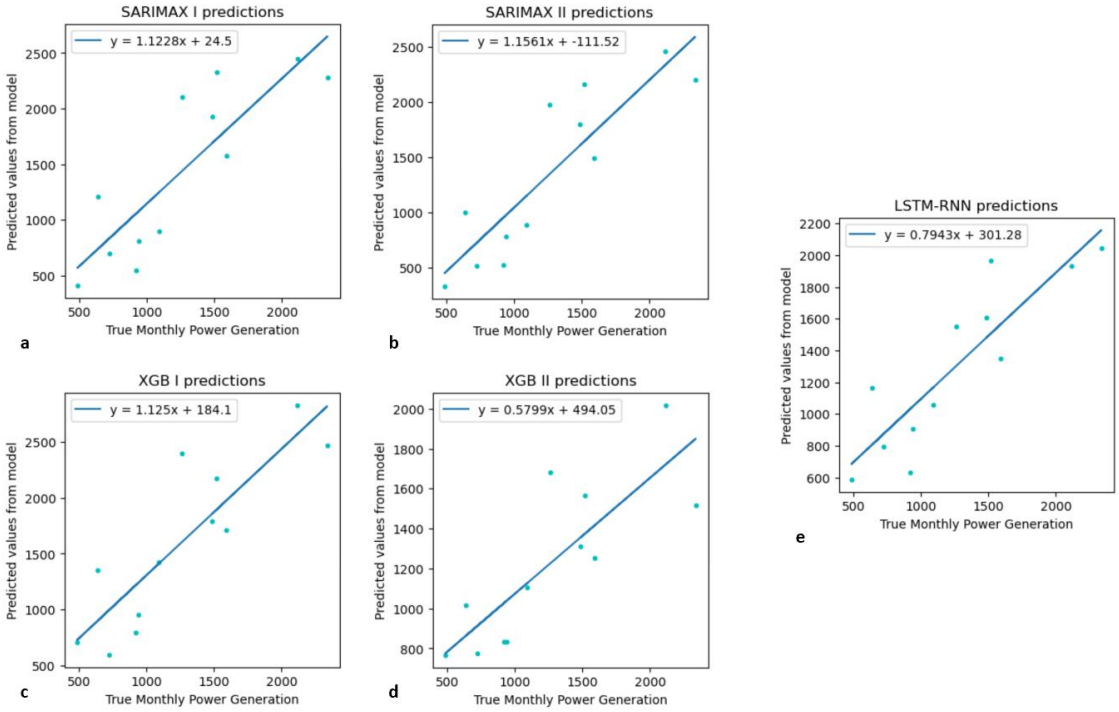

Figure 17. Plots of True power outputs verses predictions for each model and linear regression lines.

5 Discussion

Models

This project analysed how power output from a PV system changes overtime and found it to show strong seasonal patterns and a slight downward trend. All weather features were shown to have significant correlations to the power outputs with *Sun Hours* having the strongest positive correlation. The three types of models (statistical, ML and DL) were all developed using lags of the target and weather features, and long-term forecasting was performed. From the results in Table 11, Figure 16, and Figure 17, the LSTM-RNN is the best performing model overall.

Studies that compare different types of models often find a DL model to perform best (Table 1). In this project, the RMSE of the LSTM-RNN is considerably smaller than both SARIMAX models. This supports findings from studies that compare LSTM-RNN models to statistical models and found LSTM-RNNs have the best performance (Kim, Akhtar and Yang, 2023; Lee and Kim, 2019). DL models are also reported to outperform ML models (Essam *et al.*, 2022; Su, Batzelis and Pal, 2019; Sangrody *et al.*, 2018), corresponding to the findings in this project where the LSTM-RNN outperforms both XGB models across all performance metrics. This project and previous studies find LSTM-RNNs, when compared with other models (including other DL models), often have the best performance (Table 1). However, Essam *et al.* (2022) found an LSTM-RNN to be the worst performing model compared with an ANN and ML models. Therefore, some uncertainty remains surrounding the best model for PV power output forecasting.

The SARIMAX II model has the best R^2 of 0.78 followed by the LSTM-RNN (0.77). Kim, Akhtar and Yang (2023) SARIMAX model had R^2 values between 0.84-0.92. Dhaked, Dadhich and Birla (2023) and Kim, Akhtar and Yang (2023) produce LSTM-RNN models with R^2 values of 0.98 and 0.943 respectively. These values are expected to be higher than models produced in this project as they used much shorter horizon times. The LSTM-RNN model by Jung *et al.* (2020) produced a R^2 of 0.724, closer to the values produced in this project as they use a longer horizon time of one month. The R^2 for the models built in this project show good model performance considering a much long horizon time than previously studied. RMSE and MAE are harder to compare between studies due to different units of measurement used.

Implications and User Requirements

Accurate forecasting models can have numerous positive effects for the PV industry: power plants can utilise models to manage systems and prepare for potential high power production periods (Tuohy *et al.*, 2015). This is especially important with the changing climate affecting PV systems outputs. Regarding local systems, models tailored to areas could allow local grid companies to prepare for large power outputs from the residential systems. Forecasting models using data from residential systems also allow PV companies to estimate payback periods and system lifespan.

Reliable power output prediction ensures a positive opinion about PV systems from prosumers which would likely lead to increased number of PV installations on residential rooftops. Utilising these would have benefits in multiple areas such as increased renewable energy sources; decreased need for farmland or biodiverse land to be used for PV plants; and monetary benefits for prosumers and PV companies.

Limitations and Future Work

Both SARIMAX models and XGB model have similar patterns in their predictions that mimic the true values, Figure 16. All models appear to better predict winter months with low power outputs

(Figures 16 and 17)**.** However, this may be due to the test data's winter months being closer in time to the end of the training data. Subsequent work could compare different length horizon times for the LSTM-RNN and test whether the model always more accurately predicts the winter months or just the values closer to the training set.

Solar irradiance and humidity are used in other work and pollution and dust are seen to affect PV system output (Bergin *et al.*, 2017). In future work, these features could be included, feature engineering performed and measure the contributions of each feature to the predictions. This would give indications of the most important features for forecasting. However, adding features can rapidly increase the computational expense of the development process.

As LSTM has the lowest RMSE, and DL models often outperform other models, future work could further develop DL models, such as deep-RNNs, CNNs and gated recurrent units, and compare them to the LSMT-RNN model. Hybrid models are also seen to have high performance. The review by Rajagukguk, Ramadhan and Lee (2020) states LSTM models have the best performance of standalone models and a CNN-LSTM hybrid model had the best performance overall. Su, Batzelis and Pal (2019) and Vagropoulos *et al.* (2016) also show hybrid models to outperform other models. Hence, hybrid models could be developed and compared.

DL models often perform better with larger datasets (Mellit *et al.*, 2020). This dataset is small compared with some previous studies. With additional data, the LSTM-RNN performance could be increased further and the R^2 value may be improved.

6 Conclusion

This project picked three commonly used, well performing and diverse methods to forecast PV system output. These models were developed using data from a small residential PV system in the UK and used a longer horizon time than previous studies. Results were compared using plots, RMSE, MAE and R^2. The LSTM-RNN is the suggested model from the performance metrics. Future work can compare other DL models to the LSRM-RNN and evaluate different length horizon times on model accuracy.

7 REFERENCES

AlShafeey, M. and Csáki, C. (2021) Evaluating neural network and linear regression photovoltaic power forecasting models based on different input methods. *Energy Reports* [online]. 7, pp. 7601–7614. Available from: https://www.sciencedirect.com/science/article/pii/S2352484721011446 [Accessed 19 October 2023].

Altman, N. and Krzywinski, M. (2018) The curse(s) of dimensionality. *Nature Methods* [online]. 15 (6), pp. 399–400. Available from: https://www.nature.com/articles/s41592-018-0019-x [Accessed 8 December 2023].

Bergin, M.H., Ghoroi, C., Dixit, D., Schauer, J.J. and Shindell, D.T. (2017) Large Reductions in Solar Energy Production Due to Dust and Particulate Air Pollution. *Environmental Science & Technology Letters* [online]. 4 (8), pp. 339–344. Available from: https://doi.org/10.1021/acs.estlett.7b00197 [Accessed 2 November 2023].

Box, G.E.P., Jenkins, G.M. and Reinsel, G.C. (2008) *Time Series Analysis: Forecasting and Control* 4th edition. [online]. Wiley. Available from: https://learning.oreilly.com/library/view/time-series-analysis/9780470272848/ [Accessed 16 November 2023].

Chen, T. and Guestrin, C. (2016) *XGBoost: A Scalable Tree Boosting System*. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* [online]. pp. 785–794. Available from: http://arxiv.org/abs/1603.02754 [Accessed 15 November 2023].

Dhaked, D.K., Dadhich, S. and Birla, D. (2023) Power output forecasting of solar photovoltaic plant using LSTM. *Green Energy and Intelligent Transportation* [online]. 2 (5), p. 100113. Available from: https://www.sciencedirect.com/science/article/pii/S277315372300049X [Accessed 29 December 2023].

Didavi, A.B.K., Agbokpanzo, R.G. and Agbomahena, M. (2021) *Comparative study of Decision Tree, Random Forest and XGBoost performance in forecasting the power output of a photovoltaic system*. In: *2021 4th International Conference on Bio-Engineering for Smart Technologies (BioSMART)* [online]2021 4th International Conference on Bio-Engineering for Smart Technologies (BioSMART). pp. 1–5. Available from: https://ieeexplore.ieee.org/abstract/document/9677566 [Accessed 1 November 2023].

Essam, Y., Ahmed, A.N., Ramli, R., Chau, K.-W., Idris Ibrahim, M.S., Sherif, M., Sefelnasr, A. and El-Shafie, A. (2022) Investigating photovoltaic solar power output forecasting using machine learning algorithms. *Engineering Applications of Computational Fluid Mechanics* [online]. 16 (1), pp. 2002–2034. Available from: https://doi.org/10.1080/19942060.2022.2126528 [Accessed 27 December 2023].

Feron, S., Cordero, R.R., Damiani, A. and Jackson, R.B. (2021) Climate change extremes and photovoltaic power output. *Nature Sustainability* [online]. 4 (3), pp. 270–276. Available from: https://www.nature.com/articles/s41893-020-00643-w [Accessed 2 November 2023].

Fjelkestam Frederiksen, C.A. and Cai, Z. (2022) Novel machine learning approach for solar photovoltaic energy output forecast using extra-terrestrial solar irradiance. *Applied Energy* [online]. 306, p. 118152. Available from: https://www.sciencedirect.com/science/article/pii/S0306261921014276 [Accessed 6 November 2023].

Gautier, A., Jacqmin, J. and Poudou, J.-C. (2018) The prosumers and the grid. *Journal of Regulatory Economics* [online]. 53 (1), pp. 100–126. Available from: https://doi.org/10.1007/s11149-018-9350-5 [Accessed 2 January 2024].

Guan, Y. *et al.* (2023) Burden of the global energy price crisis on households. *Nature Energy* [online]. 8 (3), pp. 304–316. Available from: https://www.nature.com/articles/s41560-023-01209-8 [Accessed 1 January 2024].

Historic station data - Met Office. (2023) [online]. Available from: https://www.metoffice.gov.uk/research/climate/maps-and-data/historic-station-data [Accessed 12 December 2023].

HM Government, Department for Energy Security and Net Zero (2023) *Powering Up Britain: Energy Security Plan* [online]. Available from: https://assets.publishing.service.gov.uk/media/642708eafbe620000f17daa2/powering-up-britain-energy-security-plan.pdf [Accessed 2 January 2024].

Hodson, T.O. (2022) Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not. *Geoscientific Model Development* [online]. 15 (14), pp. 5481–5487. Available from: https://gmd.copernicus.org/articles/15/5481/2022/gmd-15-5481-2022.html [Accessed 18 December 2023].

Hu, A., Levis, S., Meehl, G.A., Han, W., Washington, W.M., Oleson, K.W., van Ruijven, B.J., He, M. and Strand, W.G. (2016) Impact of solar panels on global climate. *Nature Climate Change* [online]. 6 (3), pp. 290–294. Available from: https://www.nature.com/articles/nclimate2843 [Accessed 3 November 2023].

Hunter, J., Dale, D., Firing, E. and Droettboom, M. (2023) *Using Matplotlib — Matplotlib 3.8.2 documentation*. 2023 [online]. Available from: https://matplotlib.org/stable/users/index [Accessed 9 December 2023].

Hyndman, R. and Athanasopoulos, G. (2018) *Forecasting: Principles and Practice* 2nd edition. [online]. Melbourne, Australia, OTexts. Available from: https://otexts.com/fpp2/backshift.html [Accessed 15 November 2023].

Inman, R.H., Pedro, H.T.C. and Coimbra, C.F.M. (2013) Solar forecasting methods for renewable energy integration. *Progress in Energy and Combustion Science* [online]. 39 (6), pp. 535–576. Available from: https://www.sciencedirect.com/science/article/pii/S0360128513000294 [Accessed 28 December 2023].

Jerez, S. *et al.* (2015) The impact of climate change on photovoltaic power generation in Europe. *Nature Communications* [online]. 6 (1), p. 10014. Available from: https://www.nature.com/articles/ncomms10014 [Accessed 2 November 2023].

Jung, Y., Jung, J., Kim, B. and Han, S. (2020) Long short-term memory recurrent neural network for modeling temporal patterns in long-term power forecasting for solar PV facilities: Case study of South Korea. *Journal of Cleaner Production* [online]. 250, p. 119476. Available from: https://www.sciencedirect.com/science/article/pii/S095965261934346X [Accessed 1 November 2023].

Kim, E., Akhtar, M.S. and Yang, O.-B. (2023) Designing solar power generation output forecasting methods using time series algorithms. *Electric Power Systems Research* [online]. 216, p. 109073. Available from: https://www.sciencedirect.com/science/article/pii/S0378779622011221 [Accessed 5 October 2023].

Kingma, D.P. and Ba, J. (2015) *Adam: A Method for Stochastic Optimization* [online]. Available from: http://arxiv.org/abs/1412.6980 [Accessed 9 December 2023].

Lee, D. and Kim, K. (2019) Recurrent Neural Network-Based Hourly Prediction of Photovoltaic Power Output Using Meteorological Information. *Energies* [online]. 12 (2), p. 215. Available from: https://www.mdpi.com/1996-1073/12/2/215 [Accessed 24 December 2023].

Lipperheide, M., Bosch, J.L. and Kleissl, J. (2015) Embedded nowcasting method using cloud speed persistence for a photovoltaic power plant. *Solar Energy* [online]. 112, pp. 232–238. Available from: https://www.sciencedirect.com/science/article/pii/S0038092X1400557X [Accessed 7 November 2023].

Liu, L., He, G., Wu, M., Liu, G., Zhang, H., Chen, Y., Shen, J. and Li, S. (2023) Climate change impacts on planned supply–demand match in global wind and solar energy systems. *Nature Energy* [online].

8 (8), pp. 870–880. Available from: https://www.nature.com/articles/s41560-023-01304-w [Accessed 2 November 2023].

Mellit, A., Massi Pavan, A., Ogliari, E., Leva, S. and Lughi, V. (2020) Advanced Methods for Photovoltaic Output Power Forecasting: A Review. *Applied Sciences* [online]. 10 (2), p. 487. Available from: https://www.mdpi.com/2076-3417/10/2/487 [Accessed 1 November 2023].

Mishra, P., Pandey, C.M., Singh, U., Gupta, A., Sahu, C. and Keshri, A. (2019) Descriptive Statistics and Normality Tests for Statistical Data. *Annals of Cardiac Anaesthesia* [online]. 22 (1), pp. 67–72. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6350423/ [Accessed 4 December 2023].

pandas - Python Data Analysis Library. (2023). 2023 [online]. Available from: https://pandas.pydata.org/ [Accessed 8 December 2023].

Perktold, J. *et al.* (2023) *statsmodels/statsmodels: Release 0.14.0* [online]. Available from: https://zenodo.org/record/593847 [Accessed 16 November 2023].

Phan, Q.-T., Wu, Y.-K. and Phan, Q.-D. (2021) *Short-term Solar Power Forecasting Using XGBoost with Numerical Weather Prediction*. In: *2021 IEEE International Future Energy Electronics Conference (IFEEC)* [online]2021 IEEE International Future Energy Electronics Conference (IFEEC). pp. 1–6. Available from: https://ieeexplore.ieee.org/abstract/document/9661874 [Accessed 6 November 2023].

'Photovoltaic power station' (2023) *Wikipedia* [online]. Available from: https://en.wikipedia.org/w/index.php?title=Photovoltaic_power_station&oldid=1179207461 [Accessed 6 November 2023].

Poti, K.D., Naidoo, R.M., Mbungu, N.T. and Bansal, R.C. (2023)'Intelligent solar photovoltaic power forecasting' *Energy ReportsProceedings of 2022 7th International Conference on Renewable Energy and Conservation* [online]. 9, pp. 343–352. Available from: https://www.sciencedirect.com/science/article/pii/S2352484723012416 [Accessed 1 January 2024].

Pykes, K. (2023) *PyTorch vs TensorFlow vs Keras for Deep Learning: A Comparative Guide*. 2023 [online]. Available from: https://www.datacamp.com/tutorial/pytorch-vs-tensorflow-vs-keras [Accessed 9 December 2023].

PyTorch documentation — PyTorch 2.1 documentation. (2023). 2023 [online]. Available from: https://pytorch.org/docs/stable/index.html [Accessed 9 December 2023].

Rajagukguk, R.A., Ramadhan, R.A.A. and Lee, H.-J. (2020) A Review on Deep Learning Models for Forecasting Time Series Data of Solar Irradiance and Photovoltaic Power. *Energies* [online]. 13 (24), p. 6623. Available from: https://www.mdpi.com/1996-1073/13/24/6623 [Accessed 1 November 2023].

Ramli, M.A.M., Prasetyono, E., Wicaksana, R.W., Windarko, N.A., Sedraoui, K. and Al-Turki, Y.A. (2016) On the investigation of photovoltaic output power reduction due to dust accumulation and weather conditions. *Renewable Energy* [online]. 99, pp. 836–844. Available from: https://www.sciencedirect.com/science/article/pii/S0960148116306784 [Accessed 6 November 2023].

Raza, M.Q., Nadarajah, M. and Ekanayake, C. (2016) On recent advances in PV output power forecast. *Solar Energy* [online]. 136, pp. 125–144. Available from:

https://www.sciencedirect.com/science/article/pii/S0038092X16302547 [Accessed 7 November 2023].

Ruder, S. (2017) *An overview of gradient descent optimization algorithms* [online]. Available from: http://arxiv.org/abs/1609.04747 [Accessed 14 December 2023].

Sangrody, H., Zhou, N., Tutun, S., Khorramdel, B., Motalleb, M. and Sarailoo, M. (2018) *Long term forecasting using machine learning methods*. In: *2018 IEEE Power and Energy Conference at Illinois (PECI)* [online]2018 IEEE Power and Energy Conference at Illinois (PECI). pp. 1–5. Available from: https://ieeexplore.ieee.org/abstract/document/8334980 [Accessed 24 December 2023].

Shadid, R., Khawaja, Y., Bani-Abdullah, A., Akho-Zahieh, M. and Allahham, A. (2023) Investigation of weather conditions on the output power of various photovoltaic systems. *Renewable Energy* [online]. 217, p. 119202. Available from: https://www.sciencedirect.com/science/article/pii/S0960148123011175 [Accessed 6 November 2023].

Shafiee, S. and Topal, E. (2009) When will fossil fuel reserves be diminished? *Energy Policy* [online]. 37 (1), pp. 181–189. Available from: https://www.sciencedirect.com/science/article/pii/S0301421508004126 [Accessed 3 November 2023].

Sherstinsky, A. (2020) Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. *Physica D: Nonlinear Phenomena* [online]. 404, p. 132306. Available from: https://www.sciencedirect.com/science/article/pii/S0167278919305974 [Accessed 28 December 2023].

Shi, J., Lee, W.-J., Liu, Y., Yang, Y. and Wang, P. (2012) Forecasting Power Output of Photovoltaic Systems Based on Weather Classification and Support Vector Machines. *IEEE Transactions on Industry Applications* [online]IEEE Transactions on Industry Applications. 48 (3), pp. 1064–1069. Available from: https://ieeexplore.ieee.org/abstract/document/6168891 [Accessed 1 November 2023].

Shireen, T., Shao, C., Wang, H., Li, J., Zhang, X. and Li, M. (2018) Iterative multi-task learning for time-series modeling of solar panel PV outputs. *Applied Energy* [online]. 212, pp. 654–662. Available from: https://www.sciencedirect.com/science/article/pii/S0306261917317737 [Accessed 6 November 2023].

'sklearn.model_selection.RandomizedSearchCV' (2023) *scikit-learn*. 2023 [online]. Available from: https://scikit-learn/stable/modules/generated/sklearn.model_selection.RandomizedSearchCV.html [Accessed 16 November 2023].

Song, Z., Liu, J. and Yang, H. (2021) Air pollution and soiling implications for solar photovoltaic power generation: A comprehensive review. *Applied Energy* [online]. 298, p. 117247. Available from: https://www.sciencedirect.com/science/article/pii/S030626192100667X [Accessed 2 November 2023].

Su, D., Batzelis, E. and Pal, B. (2019) *Machine Learning Algorithms in Forecasting of Photovoltaic Power Generation*. In: *2019 International Conference on Smart Energy Systems and Technologies (SEST)* [online]2019 International Conference on Smart Energy Systems and Technologies (SEST). pp. 1–6. Available from: https://ieeexplore.ieee.org/abstract/document/8849106 [Accessed 6 November 2023].

'The Causes of Climate Change' (2023) *Climate Change: Vital Signs of the Planet*. 2023 [online]. Available from: https://climate.nasa.gov/causes [Accessed 3 November 2023].

The MCS Data Dashboard - MCS. (no date) [online]. Available from: https://datadashboard.mcscertified.com/InstallationInsights [Accessed 3 October 2023].

Tuohy, A. *et al.* (2015) Solar Forecasting: Methods, Challenges, and Performance. *IEEE Power and Energy Magazine* [online]IEEE Power and Energy Magazine. 13 (6), pp. 50–59. Available from: https://ieeexplore.ieee.org/abstract/document/7299804 [Accessed 23 December 2023].

Vagropoulos, S.I., Chouliaras, G.I., Kardakos, E.G., Simoglou, C.K. and Bakirtzis, A.G. (2016) *Comparison of SARIMAX, SARIMA, modified SARIMA and ANN-based models for short-term PV generation forecasting*. In: *2016 IEEE International Energy Conference (ENERGYCON)* [online]2016 IEEE International Energy Conference (ENERGYCON). pp. 1–6. Available from: https://ieeexplore.ieee.org/abstract/document/7514029 [Accessed 7 November 2023].

Verma, T., Tiwana, A.P.S., Reddy, C.C., Arora, V. and Devanand, P. (2016) *Data Analysis to Generate Models Based on Neural Network and Regression for Solar Power Generation Forecasting*. In: *2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS)* [online]2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS). pp. 97–100. Available from: https://ieeexplore.ieee.org/document/7877196 [Accessed 1 November 2023].

Waskom, M. (2021) seaborn: statistical data visualization. *Journal of Open Source Software* [online]. 6 (60), p. 3021. Available from: https://joss.theoj.org/papers/10.21105/joss.03021 [Accessed 9 December 2023].

XGBoost Documentation — xgboost 2.0.2 documentation. (2022). 2022 [online]. Available from: https://xgboost.readthedocs.io/en/stable/index.html [Accessed 16 November 2023].

XGBoost Parameters — xgboost 2.0.2 documentation. (2022). 2022 [online]. Available from: https://xgboost.readthedocs.io/en/stable/parameter.html [Accessed 16 November 2023].

8 Appendix A: Abbreviations

ANFIS: Adaptive Network Based Fuzzy Inference System

ANN: Artificial neural network

ARIMA: Autoregressive Integrated Moving Average

ARIMAX: ARIMA with exogenous features

ARMA: Autoregressive Moving Average

BPNN: Backpropagation neural network

DL: Deep learning

DNN: Deep neural network

DT: Decision Tree

ELM: Extreme Learning Machine

ENN: Elman neural network

GRNN: Generalized regression neural network

GRP: Gaussian Process Regression

KNN: K-nearest neighbours

Log R: Logarithmic regression

LR: Linear regression

MAE: Mean absolute error

MAPE: Mean absolute percentage error

ML: Machine learning

MRE: Mean relative error

MSE: Mean squared error

NARXNN: Non-linear Autoregressive Exogenous Neural Network

nRMSE: normalised RMSE

POD: Power output data

PR: Polynomial regression

PV: Photovoltaic

RF: Random Forest

RMSPE: Root mean squared percent error

RNN: Recurrent neural network

SARIMA: Seasonal ARIMA

SARIMAX: SARIMA with exogenous features

SVM: Support vector machine

SVR: Support vector regression

WD: Weather data

XGB: Extreme gradient boosting