

ML Project Progress Report

[Project Colab link](#)

Contents

- Abstract
- Steps Taken
- Results
 - EDA
 - Training
 - Testing

Abstract

This project addresses the critical challenge of solar power output forecasting by developing and evaluating a Long Short-Term Memory (LSTM) neural network model.

Utilising historical generation and environmental sensor data from two distinct solar plants, a comprehensive data pipeline w

Steps Taken

Data Loading and Initial Processing

- Generation and weather data for Plant 1 and Plant 2 were loaded from their respective CSV files.
- Merging plant and weather data respectively while standardising time formats
- Handling missing values

Exploratory Data Analysis (EDA) Pre-processing

- Null values and duplicate entries (based on `DATE_TIME` and `SOURCE_KEY`) were removed.
- IQR-based outlier filtering was applied to 'generating' data (where `AC_POWER` and `IRRADIATION` > 0).
- Outliers were identified but not removed as they constituted less than 20% of the generating data.
- Plotted frequency distribution graphs, correlation matrices, box plots, etc.

LSTM Model Data Preparation

- For each plant, inverter-level data was aggregated by `DATE_TIME` to create plant-level average signals.
- Data was resampled to a consistent 15-minute frequency, and missing values were filled using time-based interpolation.
- Cyclical time features (`hour_sin`, `hour_cos`) were extracted from the `DATE_TIME` index.
- Features (`AMBIENT_TEMPERATURE`, `MODULE_TEMPERATURE`, `IRRADIATION`, `hour_sin`, `hour_cos`) and the target variable (`AC_POWER`) for Plant 1 were scaled using `MinMaxScaler`.
- Time series data was transformed into sequences (windows) suitable for LSTM, using `N_LOOKBACK_STEPS` of 12 (3 hours of 15-minute intervals).
- **Train-Test Split:** The Plant 1 dataset was split into 80% training and 20% testing sets.

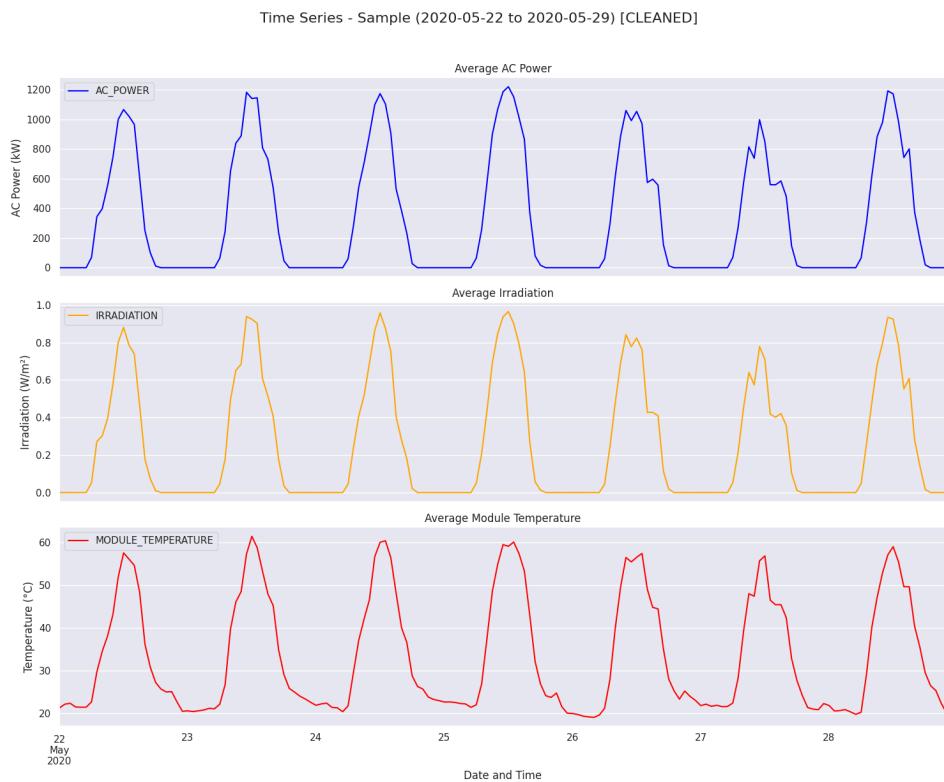
LSTM Model Building, Training, and Evaluation

- A Sequential LSTM model was built, consisting of:
 - LSTM layer (50 units, 'relu' activation)
 - Dense output layer (1 unit - correlating to target variable)

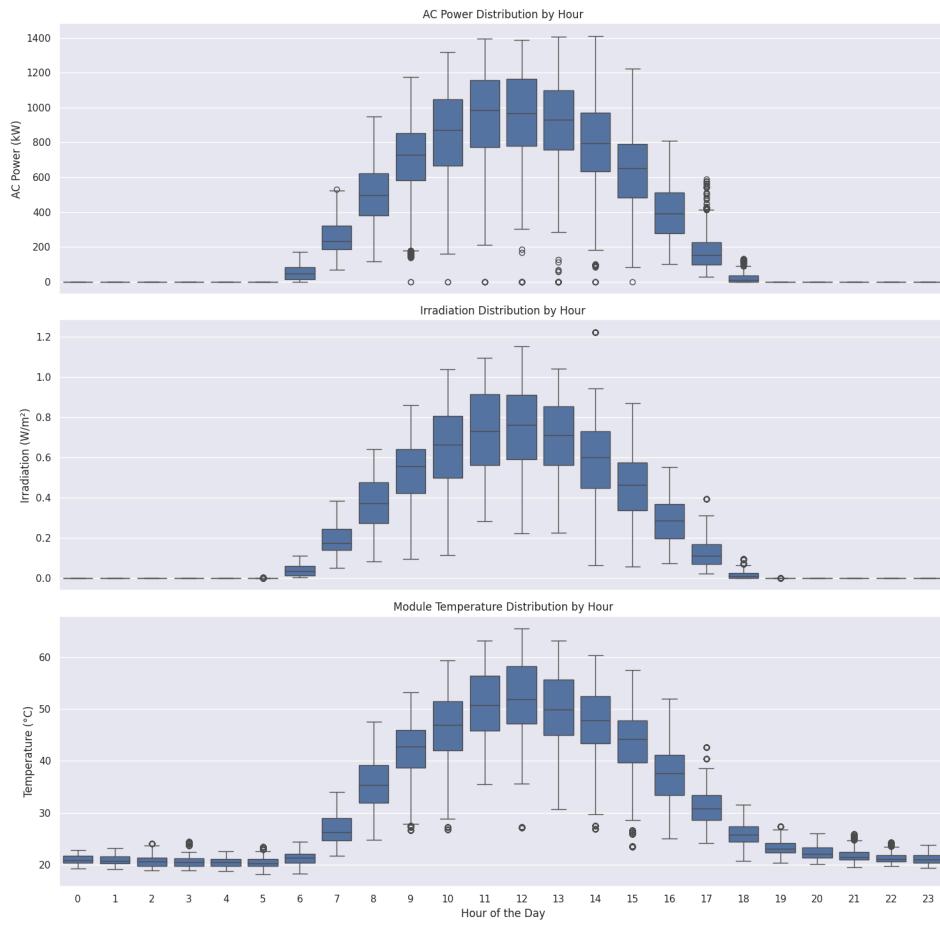
- The model was compiled using the Adam optimizer with a learning rate of 0.001 and `mean_squared_error` as the loss function.
- The LSTM model was trained on the Plant 1 training data for 50 epochs, with the Plant 1 test set used for validation.
- The trained model was evaluated on the Plant 1 test set, and the Root Mean Squared Error (RMSE) was calculated and displayed.
- Plant 2 data was loaded, preprocessed, and scaled using the same scalers fitted on Plant 1. The model's performance on this external dataset was then evaluated, and its RMSE was calculated and displayed.

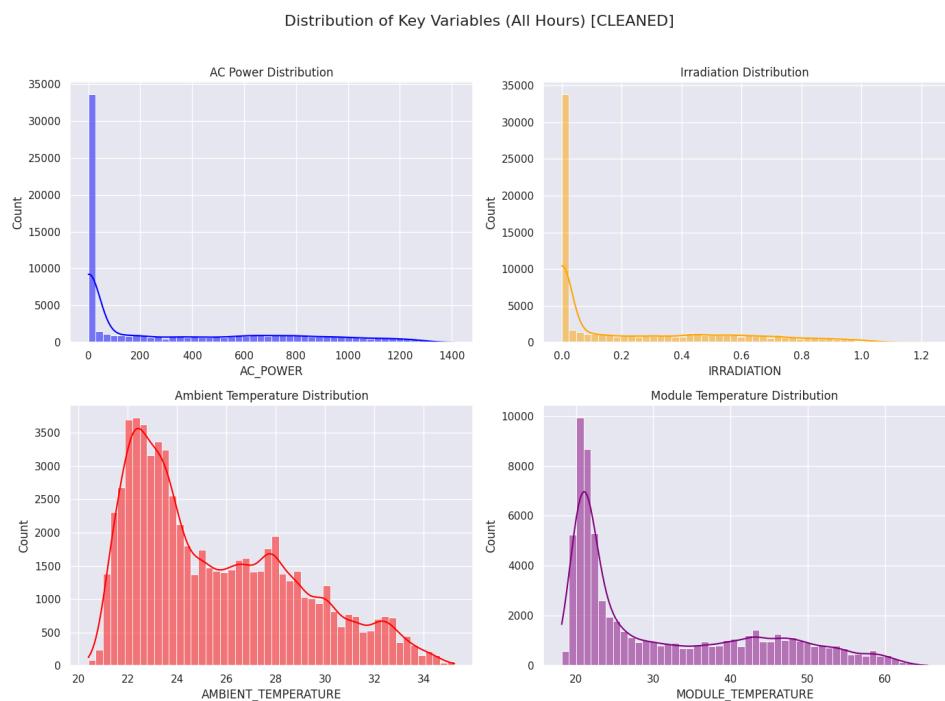
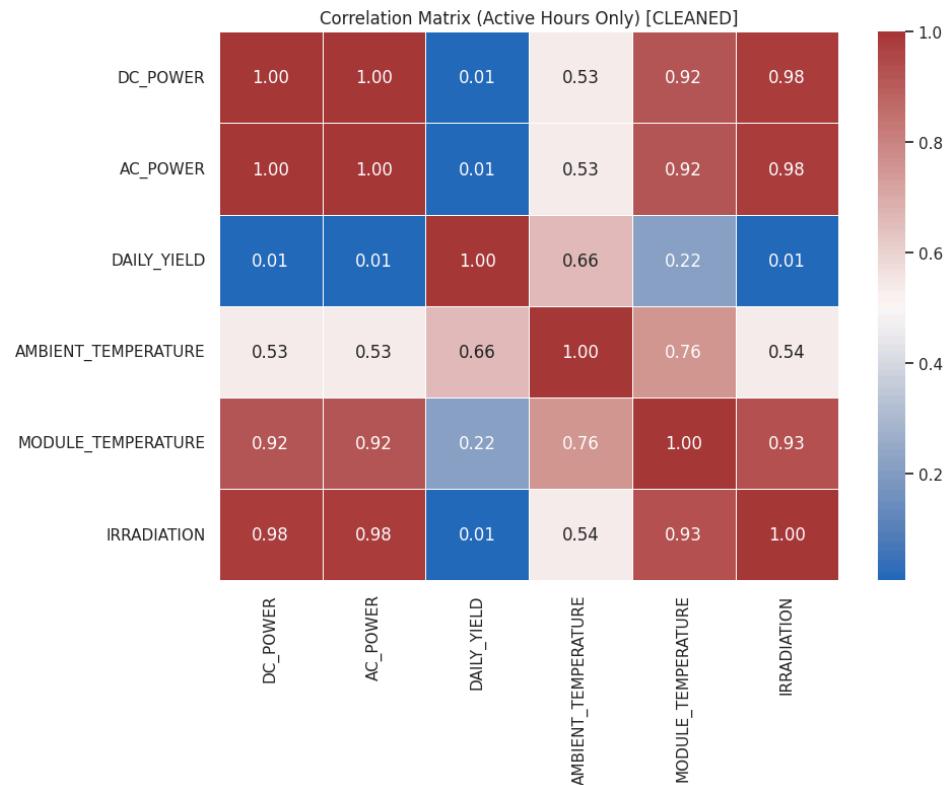
Results

EDA

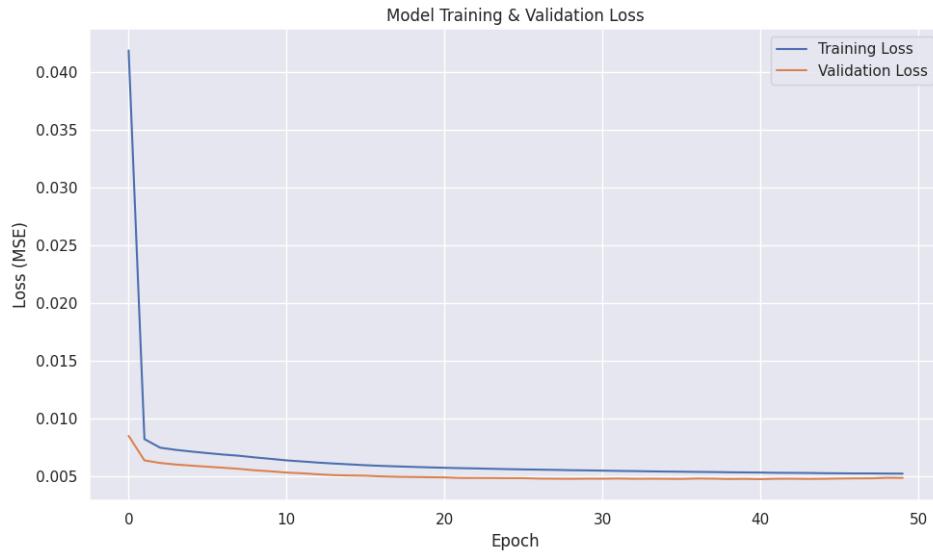


Hourly Distribution Boxplots (All Hours) [CLEANED]





Training



Testing

